Esteban Inga (Coordinator)

Digital Technology for Smart Grid

Innovative Algorithmic Solutions for Engineering Problems

Universidad Politécnica Salesiana

Digital Technology for Smart Grid. Innovative Algorithmic Solutions for Engineering Problems

Esteban Inga
(coord.)

SciELO Books / SciELO Livros / SciELO Libros

INGA, E., ed. *Digital Technology for Smart Grid*. Innovative Algorithmic Solutions for Engineering Problems [online]. Quito: Editorial Abya-Yala, 2024, 182 p. ISBN: 978-9978-10-877-2. Available from: https://books.scielo.org/id/bbw2d. https://doi.org/10.17163/abyaups.44.

All the contents of this work, except where otherwise noted, is licensed under a Creative Commons Attribution 4.0 International license.

Todo o conteúdo deste trabalho, exceto quando houver ressalva, é publicado sob a licença Creative Commons Atribição 4.0.

Todo el contenido de esta obra, excepto donde se indique lo contrario, está bajo licencia de la licencia Creative Commons Reconocimento 4.0.

Digital Technology for Smart Grid

Innovative Algorithmic Solutions for Engineering Problems

Esteban Inga (Coordinator)

Digital Technology for Smart Grid

Innovative Algorithmic Solutions for Engineering Problems

ABYA YALA

2024

Digital Technology for Smart Grid.
Innovative Algorithmic Solutions for Engineering Problems

Authors: Esteban Inga, Alexander Aguila, Diego Carrión, Wilson Pavón, Manuel Jaramillo

1st edition:
© Salesian Polytechnic University
Av, Turuhuayco 3-69 y Calle Vieja.
PDB (+593 7) 2050000
e-mail: publicaciones@ups.edu.ec
www.ups.edu.ec

MASTER OF INFORMATION AND COMMUNICATION TECHNOLOGIES FOR EDUCATION
MASTER OF ELECTRICITY
CAREER OF ELECTRICITY
RESEARCH GROUP IN INTELLIGENT ELECTRICAL GRIDS (GIREI)

Cover photo: Shutterstock

UPS ISBN: 978-9978-10-875-8
Digital ISBN: 978-9978-10-877-2
DOI: https://doi.org/10.17163/abyaups.44
Design, layout and printing: Abya-Yala University Publishing House
Circulation: 100 copies

Printed in Quito-Ecuador, January 2024
Arbitrated publication of the Salesian Polytechnic University

The content of this book is the sole responsibility of the authors

Contents

1 Presentation 7

2 Georeferencing for Planning Underground Networks . . . 11

2.1 Introduction 11

2.2 Related Works 12

2.3 Problem Formulation and Methodology 17

2.4 Matlab Coding and Results Analysis. 23

2.5 Conclusions 49

3 Multicriteria Analysis for Quality and Reliability in Electrical Systems 55

3.1 Introduction 55

3.2 Related Works 58

3.3 Problem Formulation and Methodology 61

3.4 Analysis of Results 64

3.4.1 Optimal location and dimensioning of capacitors in microgrids using a multicriteria decision algorithm 64

3.4.2 Optimal location of reclosers in electrical distribution systems considering multicriteria decision 76

3.5 Conclusions 117

4 Optimal Power Flow with Annual Demand Growth in Transmission Lines 121

4.1 Introduction 121

4.2 Configuration 124

4.3 Optimal power flows 128
4.3.1 DC Optimal power flow . . . 129
4.3.2 AC Optimal power flow . . . 130

4.4 Matlab - GAMS Coding and Results Analysis. 132
4.4.1 OPF-DC code . . . 132
4.4.2 OPF-AC code . . . 137

4.5 Conclusions 142

5 Innovative Control Paradigms for DC Motors Control . . 147

5.1 Introduction 147

5.2 Related Works 148

5.3 Methodology 150
5.3.1 First part: System Identification . . . 152
5.3.2 System identification with b = 0 . . . 154
5.3.3 Part two: PID control . . . 155
5.3.4 Part three: feedback control . . . 158

5.4 Results and Analysis 160

5.5 Conclusions 162

6 Time Series Analysis for Electrical Demand Forecasting 166

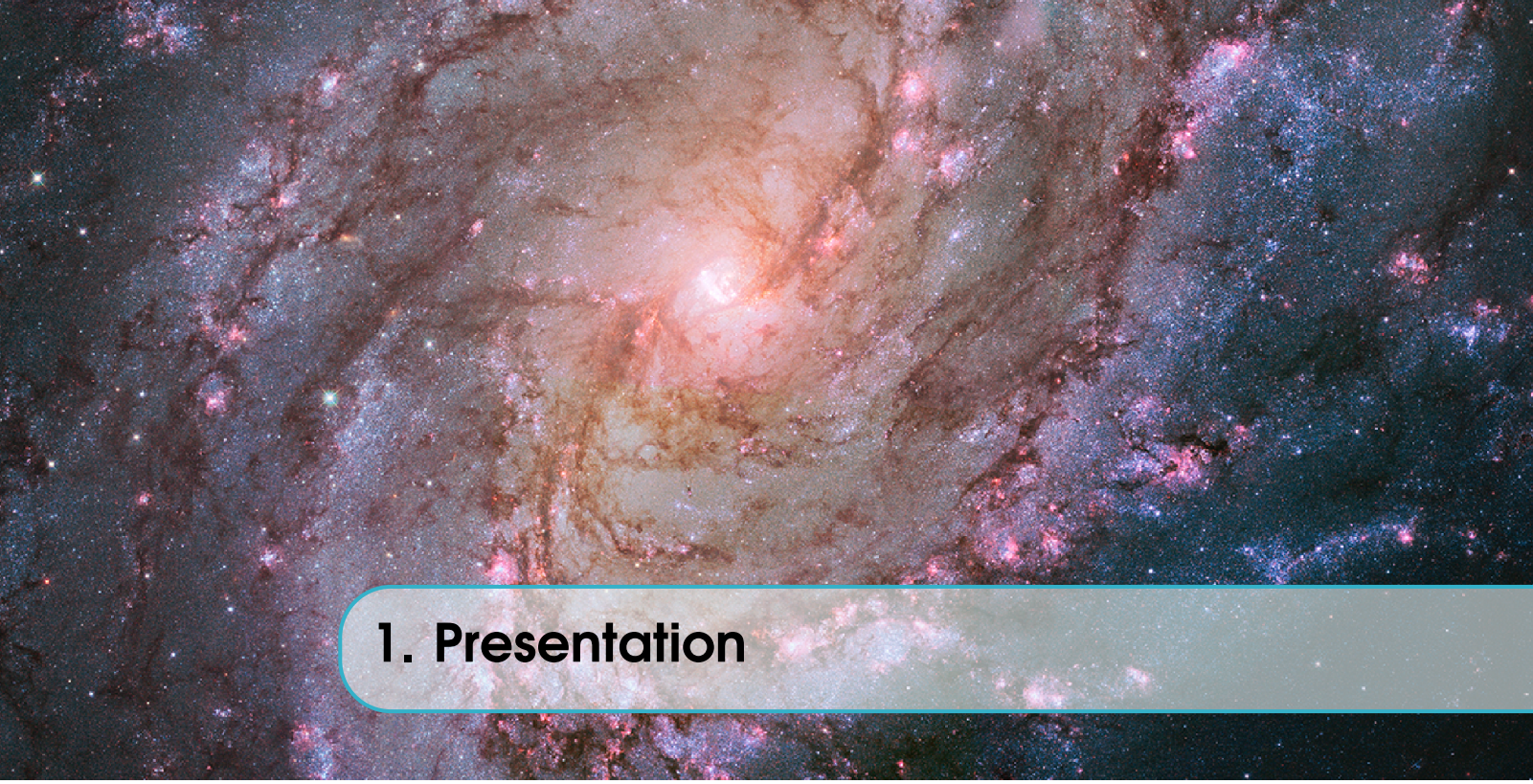
6.1 Introduction 166

6.2 Related Works 167

6.3 Problem Formulation and Methodology 171
6.3.1 Study case . . . 171
6.3.2 Time series analysis: Seasonal Auto-Regressive Integrated Moving Average (SARIMA) 172
6.3.3 Methodology for electricity consumption forecasting . . . 173

6.4 Matlab Coding and Results Analysis 173

6.5 Conclusions 179



1. Presentation

After more than ten years of the beginning of the research group in smart grids, as well as the positioning of the brand of the group known as GIREI and in which several of its members have achieved their doctoral studies, noticed the need for knowledge transfer to the classroom of undergraduate, masters and doctoral degree from their academic experiences. It is why Esteban, Alexander, Diego, Wilson, and Manuel have come together to produce this work that contains source code, pseudocode, and results that can help different engineering applications.

This book is born within the first year of deploying Artificial Intelligence (Chat-GPT, Poe, etc.), Educational Engineering, Education 5.0, and several advances in science and technology. The book proposes that the readers of the work can explore its applications and start with new research, as well as articulate results from different specialties.

The chapters on georeferencing and network planning, multi-criteria analysis, power flow optimization, DC motor control, and power demand analysis have been generated from theory to practice based on the disciplinary knowledge of the authors, the pedagogical knowledge of several years of teaching and the technological expertise that characterizes them. Then it is time to enjoy its contents:

Chapter 2 presents a valuable process for working with geo-referenced scenarios in applications such as Electrical Engineering, Traffic Engineering, Road Networks, Water Pipelines, Underground Fiber Optics, Wireless Networks, Vehicular Networks, and Electromobility. Planning models do not always imply a simulation in a dimensionless or fictitious scenario. Studies and consultancies require testing in geo-referenced environments in urban, suburban, or rural areas. In this way, it

is essential to validate the scalability and population growth of the variables that may intervene in the study. A bibliometric analysis that identifies the penetration of studies in various areas of knowledge but on geo-referenced scenarios is initially noticed for the survey. Thus, the Euclidean distance often used today is replaced by the haversine distance that includes the earth's curvature and with greater precision for specific scenarios. The substantial contribution of this work is the correction and incorporation of additional functions to those presented in MathWorks since all the study is done on Matlab, a licensed software but of great advantage for engineering research. Finally, routing work is presented as an example for wireless networks incorporating algorithms such as Dijkstra and where the connectivity matrix is evidenced when the problem is treated from the graph theory $G = (V, E)$.

Chapter 3 presents the multicriteria decision methodology as a practical approach to solving engineering problems that require an optimal joint solution. This method addresses the challenges posed by optimization problems with multiple variables and conflicting objectives or criteria. Multicriteria practices assist in identifying and evaluating a set of solutions, enabling decision-makers to select the best alternative among the contradictory criteria. The chapter describes the proposed multicriteria decision methodology for solving optimization problems in electric power distribution systems. Two practical issues are analyzed as examples: optimal reactive power compensation and optimal placement of reclosing devices. These problems involve analyzing multiple variables related to essential aspects of Electrical Engineering, such as efficiency, reliability, and power quality. The analysis includes power flow calculations and other engineering studies based on the proposed objectives. The method's effectiveness is demonstrated through its ability to handle the complexities and conflicts among the different variables. In summary, the multicriteria decision methodology is a valuable tool for solving engineering problems that require an optimal solution. Considering multiple variables and their potential conflicts, this method provides decision-makers with alternative solutions and helps them achieve the best balance among the conflicting criteria.

Chapter 4 presents an availability of interface processes between different software that allows a better study and analysis of electrical power systems, so an interface between Matlab and GAMS has been developed to solve the problem of optimal power flows, for which a generic methodology has been proposed. The designed interface allows for solving approximate (OPF-DC) and exact (OPF-AC) flows by obtaining the values of the different electrical parameters of the systems to be analyzed. The interface makes GAMS become a Matlab tool. The information processing time and the solution to the studies are less since there is only one software that executes the modeling and optimization, very similar to the processes of the specialized simulators, but with the difference that it allows

increasing variables and restrictions for the solution of the optimal power flows.

Chapter 5 presents a comprehensive strategy for controlling a DC motor, employing three different control approaches: PID control, Model Predictive Control (MPC), and state feedback control using Ackerman and Bessel polynomial strategies. The study explores the performance of each control method and its suitability for different applications. The PID control offers a reliable and widely used approach, while the MPC provides advanced predictive capabilities. The state feedback control utilizing Ackerman and Bessel polynomials demonstrates an innovative approach for achieving optimal performance. The research highlights the strengths and limitations of each strategy, emphasizing the need for further testing and validation. Overall, this study offers valuable insights into enhancing DC motor control through diverse control techniques, opening new possibilities for optimizing motor performance in various industrial applications.

Chapter 6 begins by introducing the concept of time series analysis and its importance in forecasting the future values of a process. Then focuses on electricity consumption in Quito-Ecuador from 2004 to 2018 as a specific example of a time series data set and explains the challenges associated with modeling such data, also the importance of identifying seasonality and trend in the data is emphasized, which can significantly impact the accuracy of the model. Time series modeling is explained with MATLAB code to implement each step. Then SARIMA model is presented, a widely used method for modeling time series data with seasonal patterns. Additionally, autoregression, differencing, and moving average time series are analyzed. Also, time series models are validated with the residual sum of errors RSS and average error per sample to assess the model's performance. Finally, the work demonstrates using the SARIMA model to forecast future electricity consumption values. The text highlights the importance of continuous model refinement and updating as new data becomes available. This work provides a comprehensive guide to modeling time series data using SARIMA models. Its step-by-step instructions, MATLAB code, and practical examples make it a valuable resource for anyone interested in analyzing and forecasting time series data related to electricity consumption or other variables. By following the guidelines provided in this work, readers can gain a deeper understanding of time series analysis and make more accurate predictions for future values of their data.

2. Georeferencing for Planning Underground Networks

Ph.D. Esteban Inga Ortega is Professor of the Master's Program in ICT for Education at Universidad Politécnica Salesiana (Ecuador).
Email: einga@ups.edu.ec
https://orcid.org/0000-0002-0837-0642
DOI: 10.17163/abyaups.44.347

2.1 Introduction

Often, problems related to planning electrical networks, drinking water (sewage), data, and transportation use ideal models that are not geo-referenced. However, the present work exposes a suitable and innovative methodology to achieve data analytics from OSM files. This geographic information file can be freely downloaded from $https://www.openstreetmap.org/$ (1).

Then, applying a geo-referenced model to network planning provides characteristics of the environment that are not present in a model of x and y coordinates. In this sense, using longitude and latitude allows a location with less error for distance and location calculations that are utilized for the optimal location of wireless sensors (smart meters), electrical transformers, determination of the route for burying power and data lines, as well as using the information to determine population growth in a specific area, evaluation of a rescue zone or planning geographical areas for vaccination campaigns, among other options, (2)–(4).

Over time, scientific publications seek to innovate concerning previous work by the scientific community. Thus, applications in planning electrical distribution networks based on the use of information from OSM perform modeling for the

deployment of underground networks. Simulation tools such as Cymdist (5)–(7) generally evaluate the deployed model. In this way, the aim is to serve the deployment with the minimum cost for using the electric cable and considering variables such as voltage drops (8), (9).

In addition, the location of the electric vehicle charging center or the mass public transport route evaluation will require modeling in a geo-referenced scenario capable of identifying the optimal location and considering the microscopy of vehicular traffic (10). Public transportation presents management problems; therefore, works propose models for creating and using a specific database contemplating a dynamic spatial analysis of the public transportation network to optimize public transportation schedules during peak hours (1), (11).

The problem is not knowing the location of public green spaces required for family space activities. We have proposed articulating different tools to distinguish public and private green spaces considering a Bayesian hierarchical model and OSM data from OpenStreetMap (12). Work to reduce the risk of catastrophes includes applications based on OpenStreetMap that involve sketching methods and possible flood routes (13).

This paper highlights georeferencing as a technique that integrates geographic information and analytical data for decision-making in various areas of knowledge. Using georeferencing in conjunction with data analytics allows the identification of spatial patterns and the visualization of information in a more accurate and detailed way, which can contribute to process optimization and decision-making in various fields of application. In this sense, the OpenStreetMap OSM file is a source of geographic data that can be used to generate detailed and accurate maps of different world regions. The presented figure 2.1 shows the map of a part of Turkey with the selection of dwellings obtained from a data analytics process. In addition, a feasible wireless sensor connectivity mesh and the routing between sensors considering the maximum distance constraint are presented.

Henceforth, the present article is organized as follows: Section 2 briefly reviews related work. Section 3 presents the traditional problem formulation and the methodology to solve it. Section 4 contains the results analysis and the proposed model's validation. Finally, in section 5, conclusions are presented.

2.2 Related Works

Previous works have presented an electrical network from an OSM file articulated to simulation processes. Additionally, a heuristic technique using Matlab is evidenced (14). The synthesis of geospatial data morphologically intended for the transfer and quality control of urban form is presented as a contribution of georeferencing

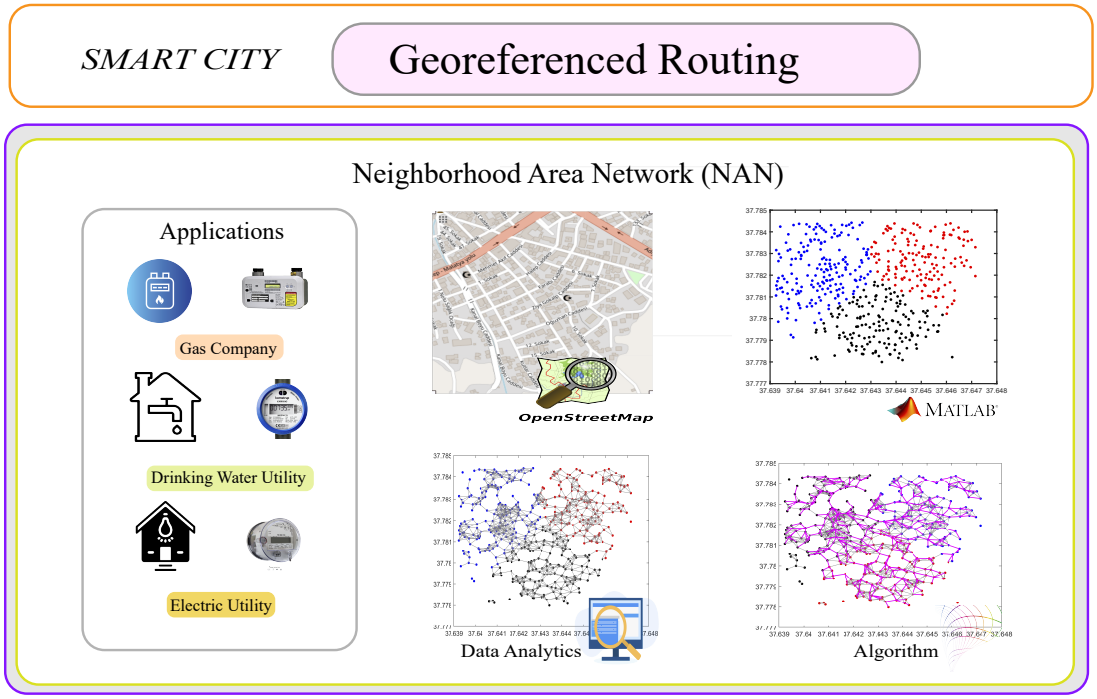


Figure 2.1: Georeferencing approach to solve problems on real scenarios

for development prediction and urban planning (15). A detailed schedule-based transit network is proposed to measure accessibility to hospitals based on transit in a growing city (16). Urban heritage management will require examination of the spatial distribution of buildings as a tangible aspect of the settlement places through spatial analysis and thus facilitate the processes and time reduction (17).

In the past, IEEE models were used to evaluate an electrical network; however, innovations involving geo-referenced scenarios are being introduced. These studies are accompanied by simulators such as Cymdist to verify that the expansion models meet scalability, reliability, resilience, and efficiency (6), (18). It has evidenced methods to detect traffic rules at intersections using GPS traces to assist location-based applications in the context of smart cities, such as accurately estimating travel time and fuel consumption from a starting point to a destination. Therefore, it proposes an automatic, fast, scalable, and inexpensive way to identify the type of intersection control, such as traffic lights and stop signs (19).

A combination of remote sensing data and statistical methods to estimate parking areas is proposed to solve the problem of adequate parking. Parking spaces and other traffic zones are detected by considering aerial images; furthermore, an obstacle model is estimated using parking zones detected from OpenStreetMap data. A relationship is found between length, street type, and parking zone ob-

tained (20), (21). The relationship between cartography and urban management provides an option for the study of trends for decision-making by professionals in charge of drawing buildings with data from OpenStreetMap (OSM) (22). The management of geo-referenced information may include information that needs to be verified due to possible failures in the on-site visit; previous works evaluate labels and filters to evidence buildings in specific areas (23). Articulating raster maps into a single system could be a non-trivial task for institutions handling geo-referenced information and requiring integration into a geographic information system (GIS) because of scarce metadata (24).

Table 2.1 presents a summary of the contributions to the use of OpenStreetMap for different georeferencing applications. The applications are various, so it was necessary to articulate a bibliometric analysis from VosViewer.

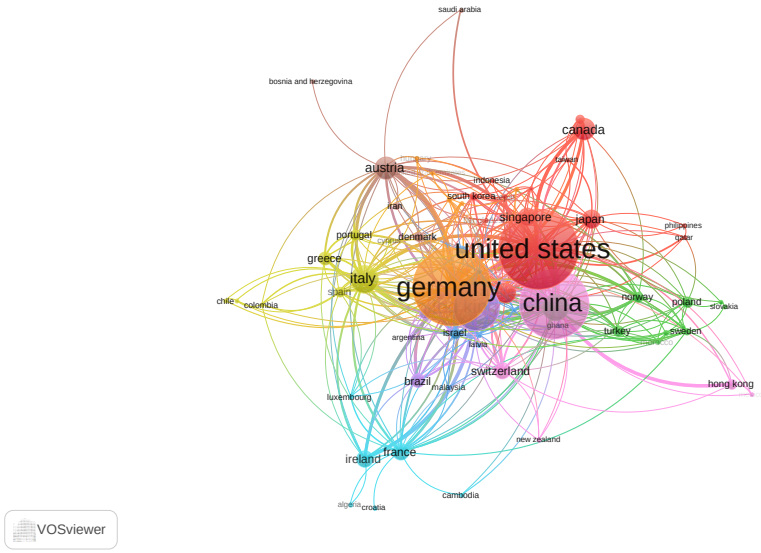
Table 2.1: Summary of related works.

		Applications					
Author, Year	Objectives	Electricity	Drinking Water	Data	Gas	Transport	Other
Garcia, 2023 [6]	Power Network Planning	✓		✓			
Kim, 2023 [16]	Healthcare accessibility			✓		✓	✓
Wu, 2023 [15]	Urban Development			✓		✓	✓
Gaugl, 2023 [14]	Power Network Planning	✓		✓			
Kersapati, 2023 [17]	Urban Management			✓		✓	✓
Kim, 2022 [25]	Small unmanned aircraft			✓			✓
Song, 2022 [23]	Remote sensing - Urban planning			✓			✓
Hacar, 2022 [22]	Urban planning			✓			✓
Hellekes, 2022 [20]	Urban Planning			✓		✓	✓
Zourlidou, 2022 [19]	Traffic Engineering		✓	✓			✓
Milleville, 2022 [24]	Gerreferencing			✓			✓
Present Work	Wireless Sensor Network	✓		✓			✓

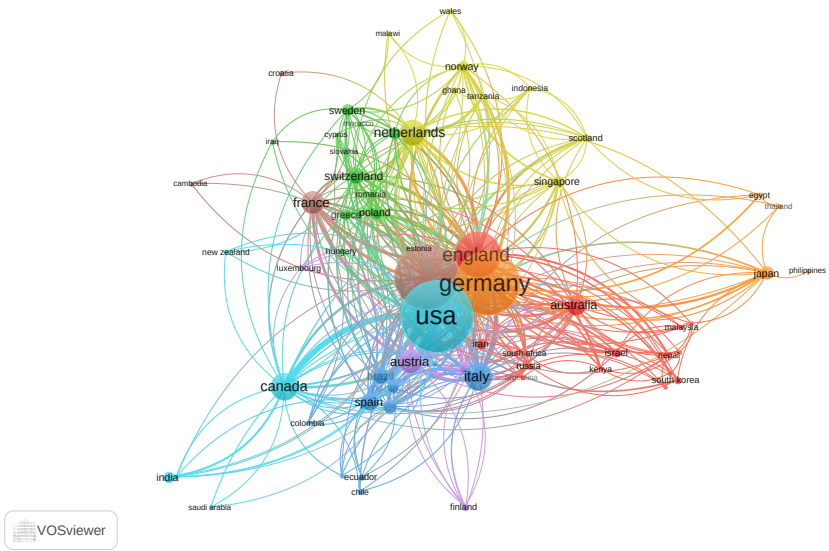
The sample used for the bibliometric analysis with VosViwer corresponds to 2000 scientific documents from Scopus and 1361 scientific documents from the Web of Science. Consequently, the bibliometric analysis corresponding to countries and authors allows us to identify the relevance concerning the amount of research using geo-referenced scenarios, accurately identifying the countries of origin and the authors that stand out. The information can be contrasted with various applications where experimentation is desired.

Figure 2.2 shows the list of countries for Scopus, among which the most crucial scientific impact are Germany, the U.S., China, the U.K., Italy, Netherlands, Austria, Canada, Ireland, and France. For Web of Science, countries such as the U.S., U.K., Germany, China, Canada, Netherlands, France, Italy, Austria, and Ireland are predominant.

Figure 2.3 shows the impact of the authors with relevance in the work of geo-referenced scenarios. Scopus authors are presented as Zipf Alexander, Boeing G, Neis P, Fan H, Jokar Arsanjani, and for Web of Science, relevant authors are Haklay, Mordechai; Zipf, Alexander; Neis, Pascal; Boeing, Geoff, Arsanjani, Jamal Jokar.

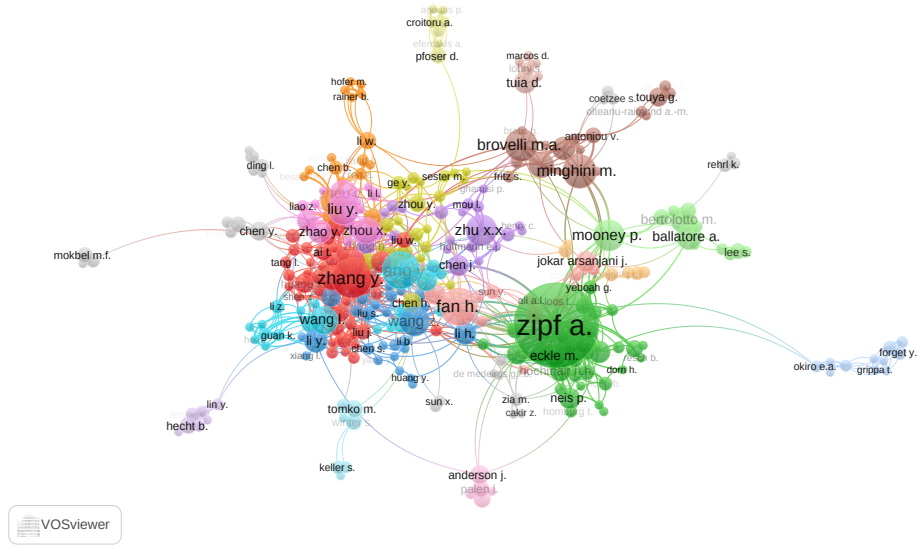


(**a**) Scopus - Countries

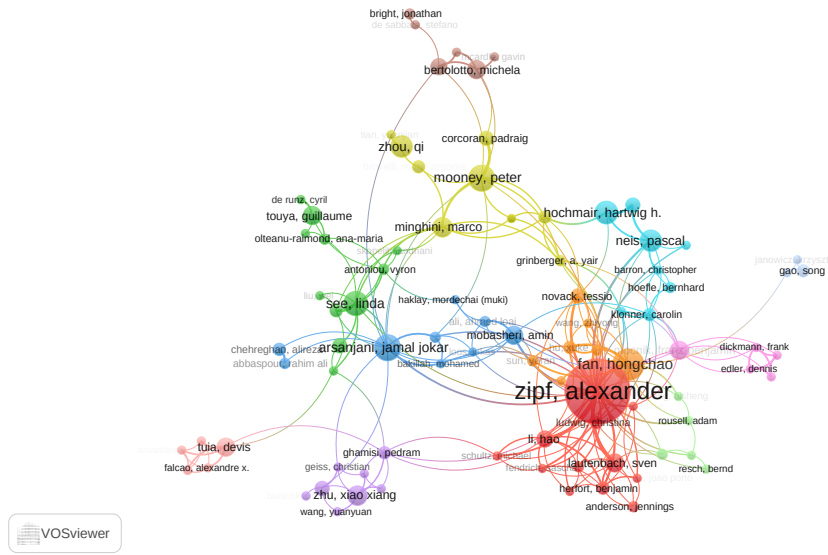


(**b**) Web of Science - Countries

Figure 2.2: Bibliometric Analysis of Countries. –(**a**) Scopus. (**b**) Web of Science. Source: Authors.



(**a**) Scopus - Authors



(**b**) Web of Science- Authors

Figure 2.3: Bibliometric Analysis of Authors. –(**a**) Scopus. (**b**) Web of Science. Source: Authors.

Using VosViewer in the bibliometric analysis of scientific documents from Web of Science and Scopus, specifically in applying OpenStreetMap in geo-referenced scenarios, allows the visualization of the scientific production of countries and authors rigorously and objectively. This tool is essential for analyzing and identifying the main trends, patterns, and research areas in this area, which translates into valuable information for strategic decision-making in developing and implementing geo-referenced projects. In summary, the graphs obtained through VosViewer provide an overview of the current state of research in this area, contributing significantly to the advancement of science and technology.

2.3 Problem Formulation and Methodology

Wireless network planning is a fundamental problem in deploying communication infrastructures, which involves making strategic decisions to reduce costs and ensure adequate coverage. In this work, a wireless network planning model is proposed based on geographic data from OpenStreetMap, which considers population growth in a scalable way and the exact location of each wireless sensor. The developed model uses an algorithm created with Matlab on a computer with an Intel Xeon 2.6GHz processor and 128GB RAM to process the data from an OSM extension file. The latitude and longitude location of the houses are used as the location of the wireless sensors.

The algorithm has as a constraint the maximum distance, dmax=45 meters, which can be modified according to the needs of the application. A feasible mesh of possible connectivity links is generated, and the internal heuristic employs Dijkstra's algorithm to find the minimum spanning tree with the distance constraint. It is important to note that there may be unconnected sensors because the maximum distance will place some sensors farther away than the allowed distance.

The proposed methodology will create an efficient and scalable wireless network planning model, reducing costs in deploying communications infrastructure and improving the quality of services offered to the population. Therefore, data analytics techniques will generate detailed maps of the region of interest to implement the proposed model. A data collection and cleaning process will be carried out to generate an OSM file containing accurate and updated information on the dwellings and existing infrastructure in the region. Subsequently, the algorithm developed for the location of wireless sensors will be used, considering the maximum distance restriction and the generation of a feasible mesh of possible connectivity links. The internal heuristics will employ Dijkstra's algorithm to find the minimum spanning tree with the distance constraint. Table 2.2 details the variables used in the equations.

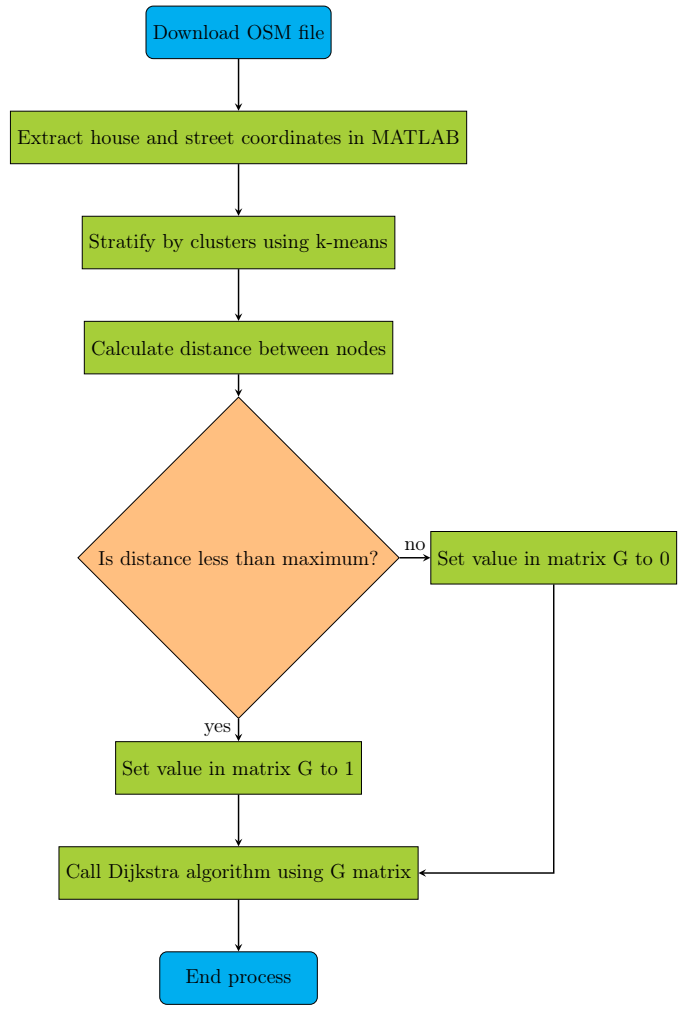

Download OSM file
Extract house and street coordinates in MATLAB
Stratify by clusters using k-means
Calculate distance between nodes
Is distance less than maximum?
no
Set value in matrix G to 0
yes
Set value in matrix G to 1
Call Dijkstra algorithm using G matrix
End process

Figure 2.4: Methodology to perform data analytics on data from an OSM file.

The Earth's shape is irregular and approximately spherical, and its surface is curved. The Haversine formula is commonly used to calculate the distance between two points on the Earth's surface accurately. This equation considers the Earth's curvature and allows for accurate distance calculations in a geo-referenced system. The distance results are usually reported in kilometers. Equations 2.1 and 2.2 provide the mathematical expressions for calculating distances using the Haversine formula for two sets of coordinates.

Table 2.2: Variables related to Haversine equation

Symbol	**Description**
D_{ij}	Distance matrix nxn - km
Ra	Earth curvature - km
lat, lon	Latitude and longitude
$distH_{ij}$	Haversine distance - km
E	Haversine Equation

$$D_{ij} = 2 * Ra * asin\sqrt{E} \tag{2.1}$$

$$E = sin^2\left(\frac{\Delta lat}{2}\right)^2 + cos\,(lat1) * cos\,(lat2) * sin^2\left(\frac{\Delta lon}{2}\right)^2 \tag{2.2}$$

This paper presents the possibility of performing data mining on an OSM file to determine the type of information available and how it could be used in basic, applied, and quantitative research. Mathworks presents a set of functions that serve for a process in Matlab, source: https://la.mathworks.com/matlabcentral/fileexchange/35819-openstreetmap-functions, and they are the following:

- assign _from _parsed.m
- debug_openstreetmap.m
- extract _connectivity.m
- get_unique_node _xy.m
- get_way_tag_key.m
- load_osm_xml.m
- main_mapping.m
- parse_openstreetmap.m
- parse_osm.m
- plot_nodes
- plot_road_network.m

- plot_route.m
- plot_way.m
- route_planner.m
- show_map.m
- usage_example.m

However, more than the exposed functions are required because they generate errors. After all, new functions are requested in Matlab that can be downloaded from the Internet by searching for them with the following names:

- xml2struct_fex28518.m
- lat_lon_proportions.m
- takehold.m
- plotmd.m
- textmd.m
- restorehold.m
- givehold.m

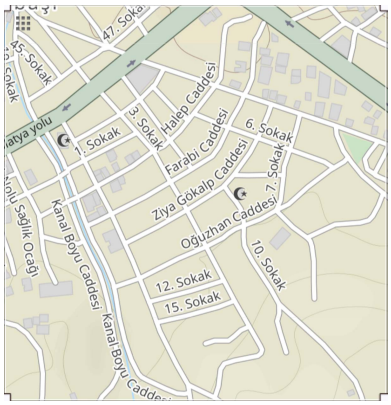
The features of each OpenStreetMap layer are detailed below:

- Standard layer: The Standard layer is the default base map layer of OpenStreetMap, which contains basic information such as roads, buildings, water bodies, and administrative boundaries. This layer is designed to provide a general view of a specific area and is suitable for navigation and available mapping applications.
- CyclOSM layer: The CyclOSM layer is specifically designed for cyclists and displays relevant information, such as bike routes, bike lanes, and bike parking facilities. It also includes information on points of interest for cyclists, such as bike shops and repair facilities.
- Cyclist Map layer: Similar to the CyclOSM layer, the Cyclist Map layer is designed for cyclists and displays information specific to them, such as bike routes, bike lanes, and bike parking facilities. It also focuses on safety and displays areas where cyclists should exercise caution.
- Transportation Map layer: The Transportation Map layer is designed to display information about public transportation, such as bus routes, train and metro stations, and bus stops. It may also include information about parking facilities and nearby points of interest.
- ÖPNVKarte layer: Similar to the Transportation Map layer, the ÖPNVKarte layer focuses explicitly on public transportation in Germany and displays information about bus routes, trams, metro, and trains. It may also include information about nearby points of interest.
- Humanitarian layer: The Humanitarian layer is designed for use in humanitarian crises such as natural disasters or armed conflicts. It contains relevant information for humanitarian aid, such as the location of shelters, hospitals, and water stations. It may also include information about roads and evacuation routes.

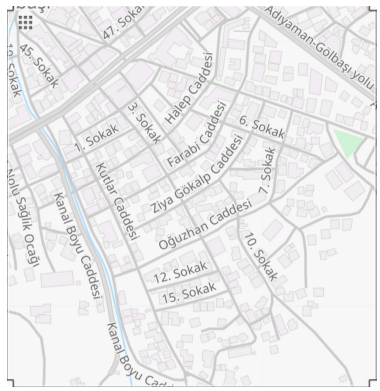
Table 2.3 presents the parameters used in the proposed georeferencing application scenario. Each variable will be modified according to the values retrieved from the OSM file. If the selected area is vast, the information on houses and streets will be increased. However, the computational time will also increase. Figure 2.5 shows the characteristic of the layers available in OpenStreetMap.

(a) Standard Layer

(b) CyclOSM Layer

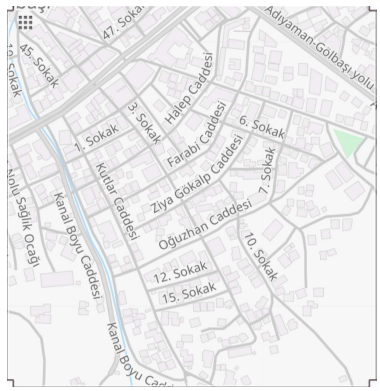


(c) Cyclist Map Layer



(d) Transportation Map Layer

(e) ÖPNVKarte Layer



(f) Humanitarian Layer

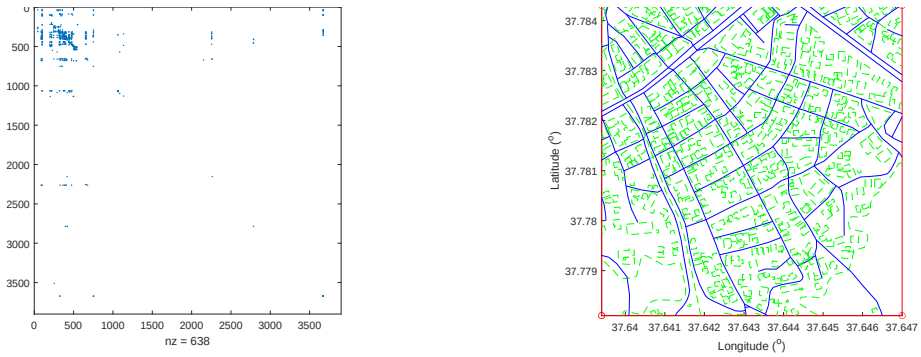
Figure 2.5: OpenStreetMap Layers Comparison: Standard, CyclOSM, Cyclist Map, Transportation Map, ÖPNVKarte, and Humanitarian

Table 2.3: Simulation parameters.

Description	Details
Location map	Ziya Gökalp Caddesi, Gölbaşı, Adıyaman, Southeastern Anatolia Region, 02500, Turkey
Latitude	[37.7781 37.7844] - Bottom to top
Longitude	[37.6393 37.6469] - Left to right.
User household	497
Geographic area	503.705,95 m^2
Coverage distance	0.045 km - constraint
Streets Sets	67
Topology evaluated	Tree
IEEE 802.15.4 Technology	Topology type Star, Tree
Candidate sites location	Location of the houses

2.4 Matlab Coding and Results Analysis.

Based on the data available in an OpenStreetMap OSM file, it is necessary to exemplify a specific application considering a geo-referenced scenario. Therefore, once all the functions are in the same folder and $*.osm$ file, you can run the Matlab script to get the first approach to map retrieval in Matlab. Figure 2.6 a) shows a graph with the connectivity matrix, with the area's connections in blue and the non-connectivity of streets or avenues in white. The figure shows the initial behavior of the articulation between Matlab and OpenStreetMap. It can be seen that figure 2.6 b) has text that does not allow us to appreciate the map.



(a) No Zeros chart (b) Stage map

Figure 2.6: The original scenario articulated Matlab & OpenStreetMap considering the functions available in Mathworks and modified them with the required new functions

The script has been modified from the original Mathworks version authored by Ioannis Filippidis in 2010 and is presented below.

```
1 clc; clear all; close all;
2 warning('off','all');
```

```
3  %====================================
4  openstreetmap_filename = 'turkey.osm';%Scenario (Chosen City)
5  [parsed_osm, osm_xml] = parse_openstreetmap(openstreetmap_filename);
6  %Retrieve OSM information
7  %Connectivity Matrix and Intersections
8  [connectivity_matrix, intersection_node_indices] =
       extract_connectivity(parsed_osm);
9  %Clean duplicate data
10 intersection_nodes = get_unique_node_xy(parsed_osm,
       intersection_node_indices);%
11 start = 1; % node global index
12 target = 9;
13 dg = or(connectivity_matrix, connectivity_matrix.'); %sparse matrix
14 [route, dist] = route_planner(dg, start, target);
15 fig = figure;
16 ax = axes('Parent', fig);%Axis
17 hold(ax, 'on')%Hold
18 plot_way(ax, parsed_osm)
19 plot_route(ax, route, parsed_osm)
20 only_nodes = 1:10:10000; % Alert! not all nodes, to reduce graphics
       memory & clutter
21 plot_nodes(ax, parsed_osm, only_nodes)
22 %====================================
23 % Page setup before printing the figure in PDF format
24 figure(1);
25 hold(ax, 'off')
26 h=gcf;
27 set(h,'PaperPositionMode','auto');
28 set(h,'PaperType','A4');
29 set(h,'PaperOrientation','landscape');
30 set(h,'Position',[10 0 500 800]);
31 set(h, 'InvertHardcopy', 'off')
32 fig = gcf;
33 fig.Color = 'white';
34 print -dpdf -r800 figure1_12
35 %====================================
36 figure(2);
37 hold(ax, 'off')
38 h=gcf;
39 set(h,'PaperPositionMode','auto');
40 set(h,'PaperType','A4');
41 set(h,'PaperOrientation','landscape');
42 set(h,'Position',[10 0 500 800]);
43 set(h, 'InvertHardcopy', 'off')
44 fig = gcf;
45 fig.Color = 'white';
46 print -dpdf -r800 figure1_13
```

Next, the code is modified to include the background image in PNG format. It is important to note that the capture of the photo image must have been previously captured and saved from OpenStreetMap. After line 4, insert the next code:

- map_map img_filename = 'figure1_standard.png';

Afterward, line 19 should be modified by the following code

- plot_way(ax, parsed_osm,map_img_filename)

Once the above changes have been made, it can be seen in the figure 2.7 that the png figure is at the bottom of the road and housing map. The figure shows the houses in green color found in the OSM file. The blue color indicates the main streets of the selected area. It is important to note that not all regions are complete and should be observed in the standard layer if the map has houses.

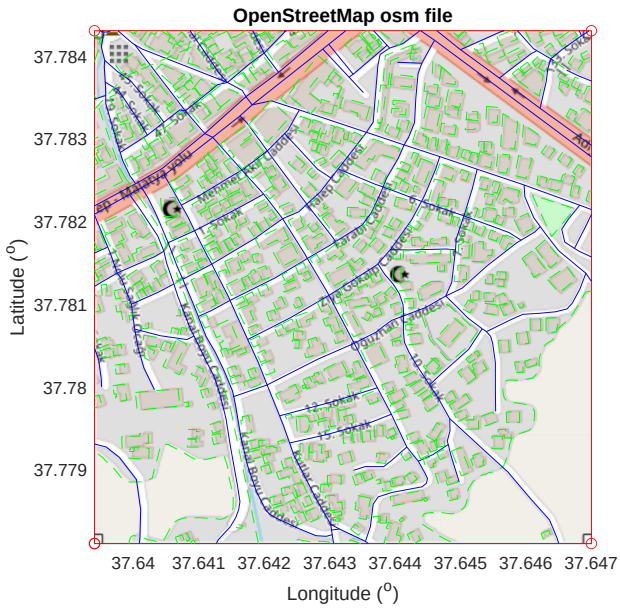


Figure 2.7: Scenario with the background image of the selected map.

The next step will be to modify the plot_way.m function to identify the information about the houses and roads. For this purpose, colors are chosen to identify each set of data. The new code for the plot_way.m function is detailed below.

```
%plot_way.m function
function [] = plot_way(ax, parsed_osm, map_img_filename)
if nargin < 3
    map_img_filename = [];
end
[bounds, node, way, ~] = assign_from_parsed(parsed_osm);
disp_info(bounds, size(node.id, 2), size(way.id, 2))
show_ways(ax, bounds, node, way, map_img_filename);

function [] = show_ways(hax, bounds, node, way, map_img_filename)
show_map(hax, bounds, map_img_filename)
house=[];zi=1;zj=1;zk=1;House=[];
key_catalog = {};
for i=1:size(way.id, 2)
    [key, val] = get_way_tag_key(way.tag{1,i} );
    % find unique way types
    if isempty(key)
    elseif isempty( find(ismember(key_catalog, key) == 1, 1) )
        key_catalog(1, end+1) = {key};
    end
    % way = highway or amenity ?
    flag = 0;
    switch key
        case 'highway'
            flag = 1;
            % bus stop ?
            if strcmp(val, 'bus_stop')
                disp('Bus stop found')
            end
        case 'amenity'
            % bus station ?
            if strcmp(val, 'bus_station')
                disp('Bus station found')
            end
            %===========================
        case 'building'
            % houses
            flag = 2;
            if strcmp(val, 'yes')
                disp('House')
            end
        case 'alt_name'
            % houses
            flag = 3;
            if strcmp(val, 'yes')
                disp('Extra Via')
            end
            %===========================
        otherwise
            disp('way without tag.')
    end
    % plot highway
```

```
53      way_nd_ids = way.nd{1, i};
54      num_nd = size(way_nd_ids, 2);
55      nd_coor = zeros(2, num_nd);
56      nd_ids = node.id;
57      for j=1:num_nd
58          cur_nd_id = way_nd_ids(1, j);
59          if ~isempty(node.xy(:, cur_nd_id == nd_ids))
60              nd_coor(:, j) = node.xy(:, cur_nd_id == nd_ids);
61          end
62      end
63      % remove zeros
64      nd_coor(any(nd_coor==0,2),:)=[];
65      if ~isempty(nd_coor)
66          % plot way (highway = blue, other = green)
67          if flag == 1
68              plot(hax, nd_coor(1,:), nd_coor(2,:), '-','LineWidth',1,
                    'color',[0.74 0.33 0.18])% plot streets
69              streets{1,zj}=[nd_coor(1,:); nd_coor(2,:)];
70              zj=zj+1;
71          else
72          end
73          if flag == 2
74              plot(hax, nd_coor(1,:), nd_coor(2,:), '-','LineWidth',1,
                    'color',[0.37 0.41 0.62]);
75              plot(hax, nd_coor(1,end), nd_coor(2,end), '<','
                    markersize',3,'color',[128/255 64/255 64/255],'
                    markerfacecolor',[255/255 127/255 39/255]);
76              house(zi,:)=[nd_coor(1,end) nd_coor(2,end)];
77              House{1,zi}=[nd_coor(1,:); nd_coor(2,:)];
78              zi=zi+1;
79          end
80          if flag == 3
81              plot(hax, nd_coor(1,:), nd_coor(2,:), '-','LineWidth',1,
                    'color',[0.74 0.33 0.18]);
82              streets2{1,zk}=[nd_coor(1,:); nd_coor(2,:)];
83              zk=zk+1;
84          end
85      end
86      %waitforbuttonpress
87  end
88  disp(key_catalog.')
89
90  function [] = disp_info(bounds, Nnode, Nway)
91  disp( ['Bounds: xmin = ' num2str(bounds(1,1)),...
92      ', xmax = ', num2str(bounds(1,2)),...
93      ', ymin = ', num2str(bounds(2,1)),...
94      ', ymax = ', num2str(bounds(2,2)) ] )
95  disp( ['Number of nodes: ' num2str(Nnode)] )
96  disp( ['Number of ways: ' num2str(Nway)] )
```

In addition, to remove the title of the figures, the function show_map.m must be entered, and the last line of the Matlab code must be disabled at the end of the

line %title(ax, 'OpenStreetMap osm file').

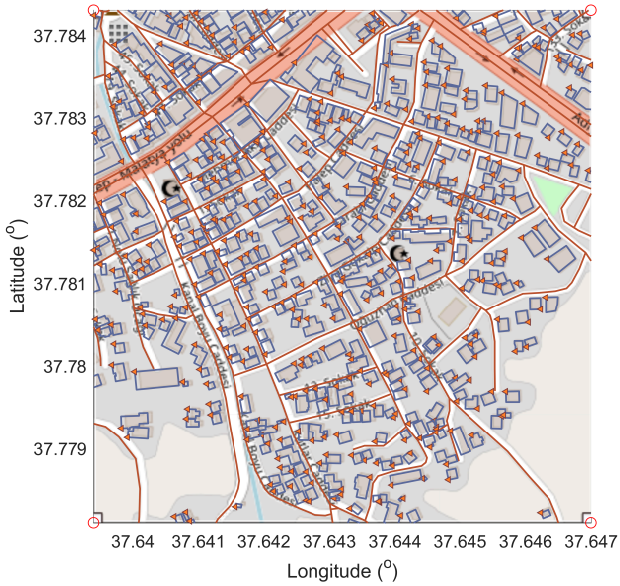


Figure 2.8: The scenario considers dwellings, roads, and the point of the residence closest to a street.

Generally, it is required to work with the information separately according to the application to be developed. The main algorithm retrieves information on houses, roads, and recreational areas. A triangle is placed at the point of each home. In this sense, the main algorithm and the plot_way function are modified and are now called new_plot_way.

The MATLAB code starts by clearing the console (clc), clearing all variables (clear all), and closing all open figures (close all). Then, you set the name of the map file to use (openstreetmap_filename) and the map image (map_img_filename). Next, the OSM (OpenStreetMap) information is extracted from the file, and the connectivity matrix and intersection node indices are obtained. Duplicate data is removed, and the start and end nodes are established.

The shortest path between these nodes is then found using Dijkstra's algorithm, and a figure is created to show the map and route. The working area is set, and data not in the specified region is removed. A legend is set for the figure, and the page is set up before printing the figure in PDF format. In summary, the code processes OSM data and displays the shortest route between two nodes on a map, which can be helpful for transportation network analysis and urban planning.

```
1 clc; clear all; close all;
2 warning('off','all');
```

```
%====================================
openstreetmap_filename = 'turkey.osm';%Scenario (Chosen City)
%Image
map_img_filename = 'figure1_estandar.png';%Imagen PNG o EPS
[parsed_osm, osm_xml] = parse_openstreetmap(openstreetmap_filename);
   %Retrieve OSM information
%Connectivity Matrix and Intersections
[connectivity_matrix, intersection_node_indices] =
   extract_connectivity(parsed_osm);
%Clean duplicate data
intersection_nodes = get_unique_node_xy(parsed_osm,
   intersection_node_indices);%
start = 1; % node global index
target = 9;
dg = or(connectivity_matrix, connectivity_matrix.'); %sparse matrix
[route, dist] = route_planner(dg, start, target);
fig = figure;
ax = axes('Parent', fig);%Axis
hold(ax, 'on')%Hold
new_plot_way(ax, parsed_osm,map_img_filename)%Include Image
plot_route(ax, route, parsed_osm)
only_nodes = 1:10:10000; % Alert! not all nodes, to reduce graphics
   memory & clutter
plot_nodes(ax, parsed_osm, only_nodes)
%====================================
% geo-referenced Scenario Work Area Grid
lonlim=[37.6393 37.6469]; % Left-Right X Limits
latlim=[37.7781 37.7844]; % Lower-Upper Y Limits
%====================================
% Vector retrieved from OSM information
[House,house,streets,z1,z2,z3,z4]=new_plot_way(ax, parsed_osm,
   map_img_filename);
[House]=delete_data(lonlim,latlim, House');
House=House';
streets=[streets];
legend([z1,z2,z3,z4],'Street','Recreation','House','Reference','
   fontname','times new roman','fontsize',13,'location','SO','
   orientation','horizontal');
% Page setup before printing the figure in PDF format
figure(1);
hold(ax, 'off'),box('on');
h=gcf;
set(h,'PaperPositionMode','auto');
set(h,'PaperType','A4');
set(h,'PaperOrientation','landscape');
set(h,'Position',[10 0 500 800]);
set(h, 'InvertHardcopy', 'off')
fig = gcf;
fig.Color = 'white';
print -dpdf -r800 figure1_14
figure(2);
hold(ax, 'off'),box('on');
```

```
48 h=gcf;
49 set(h,'PaperPositionMode','auto');
50 set(h,'PaperType','A4');
51 set(h,'PaperOrientation','landscape');
52 set(h,'Position',[10 0 500 800]);
53 set(h, 'InvertHardcopy', 'off')
54 fig = gcf;
55 fig.Color = 'white';
56 print -dpdf -r800 figure1_17
```

The MATLAB algorithm "new_plot_way" is a function that receives three input arguments: the first argument is the "ax" object representing the coordinate system in which the map will be drawn, the second argument is the "parsed_osm" object representing the map data file in OSM format. The third argument is the filename of the map image. This function uses the "assign_from_parsed" function to extract node and path information from the OSM file and then calls the "show_ways" function to draw the nodes and paths in the "ax" object. The "show_ways" function uses the "show_map" function to display the map image in the "ax" object.

Then, the "show_ways" function traverses the paths in the "way" entity to determine their type and draws them in the "ax" object with a different color for each class. If a path is a house, it also draws a marker at the last position of the path and stores the place in a "house" array. Finally, the function "new_plot_way" returns the objects "house," "house," "streets," "z1", "z2", "z3," and "z4," containing information about the drawn nodes and paths.

The algorithm demonstrates a practical application of geospatial data processing and map visualization in MATLAB. The implementation of the algorithm uses structured programming techniques. The use comprises control structures such as cycles, case selection, and nested functions that perform specific tasks. Array indexing is also used, and attention is paid to code efficiency to avoid redundancy in data representation and improve performance. The algorithm is easily understandable and modular, facilitating its maintenance and extension. Table 2.4 presents the variables used in the algorithm's 1 pseudocode, and the Matlab code is also presented below.

Table 2.4: Variables related to the new_plot_way algorithm

Name	**Description**
ax	Axis object of the plot
parsed_osm	Parsed OpenStreetMap data
map_img_filename	Filename of the map image
bounds	Boundary coordinates of the plot
node	Node coordinates
way	Way information
House	List of house coordinates
house	Temporary list of house coordinates
streets	List of street coordinates
key_catalog	Catalog of unique way tags
i	Loop index variable
key	Current way tag key
val	Current way tag value
flag	Flag variable used for differentiating between different types of ways
nd_coor	Coordinate of the current node

Algorithm 1 Function new_plot_way

```
1: function SHOW_WAYS(hax, bounds, node, way, map_img_filename)
2:     show_map(hax, bounds, map_img_filename)
3:     house ← []
4:     House ←
5:     streets ←
6:     key_catalog ←
7:     for i ← 1 to size(way.id, 2) do
8:         key, val ← get_way_tag_key(way.tag1, i)
9:         if ∼ isempty(key) & isempty(find(ismember(key_catalog, key) == 1, 1)) then
10:            key_catalogend + 1 ← key
11:        end if
12:        if key ==′ highway′ then
13:            if strcmp(val,′ bus_stop′) then
14:                continue
15:            else
16:                flag ← 1
17:                streetsend + 1 ← node.xy(:, way.nd1, i)
18:            end if
19:        end if
20:        if key ==′ amenity′ then
21:            if strcmp(val,′ bus_station′) then
22:                continue
23:            else
24:                flag ← 2
25:            end if
26:        end if
27:        if key ==′ building′ then
28:            if strcmp(val,′ yes′) then
29:                flag ← 3
30:                nd_coor ← node.xy(:, way.nd1, i)
31:                house(end + 1, :) ← [nd_coor(1, end), nd_coor(2, end)]
32:                Houseend + 1 ← nd_coor
33:            else
34:                continue
35:            end if
36:        end if
37:        if key ==′ alt_name′ then
38:            if strcmp(val,′ yes′) then
39:                flag ← 4
40:                streetsend + 1 ← node.xy(:, way.nd1, i)
41:            else
42:                continue
43:            end if
44:        end if
45:    end for
46: end function
```

```
1 function [House,house,streets,z1,z2,z3,z4] = new_plot_way(ax,
      parsed_osm, map_img_filename)
2 if nargin < 3
```

```
3       map_img_filename = [];
4       house=[];House=[];streets=[];streets2=[];z1=[];z2=[];z3=[];z4
            =[];
5   end
6   [bounds, node, way, ~] = assign_from_parsed(parsed_osm);
7   % disp_info(bounds, size(node.id, 2), size(way.id, 2))
8   [House,house,streets,z1,z2,z3,z4]=show_ways(ax, bounds, node, way,
        map_img_filename);
9   function [House,house,streets,z1,z2,z3,z4] = show_ways(hax, bounds,
        node, way, map_img_filename)
10  show_map(hax, bounds, map_img_filename)
11  house=[];zi=1;zj=1;zk=1;House=[];
12  % plot(node.xy(1,:), node.xy(2,:), '.r','markersize',10);
13  key_catalog = {};
14  for i=1:size(way.id, 2)
15      [key, val] = get_way_tag_key(way.tag{1,i} );
16      % find unique way types
17      if isempty(key)
18          %
19      elseif isempty( find(ismember(key_catalog, key) == 1, 1) )
20          key_catalog(1, end+1) = {key};
21      end
22      % way = highway or amenity ?
23      flag = 0;
24      switch key
25          case 'highway'
26              flag = 1;
27              % bus stop ?
28              if strcmp(val, 'bus_stop')
29                  disp('Bus stop found')
30              end
31          case 'amenity'
32              % bus station ?
33              flag = 2;
34              if strcmp(val, 'bus_station')
35                  disp('Bus station found')
36              end
37          case 'building'
38              % houses
39              flag = 3;
40              if strcmp(val, 'yes')
41                  disp('I found a house')
42              end
43          case 'alt_name'
44              % houses
45              flag = 4;
46              if strcmp(val, 'yes')
47                  disp('Extra Via')
48              end
49          otherwise
50              %               disp('Path.')
51      end
```

```
52      %plot highway
53      way_nd_ids = way.nd{1, i};
54      num_nd = size(way_nd_ids, 2);
55      nd_coor = zeros(2, num_nd);
56      nd_ids = node.id;
57      for j=1:num_nd
58          cur_nd_id = way_nd_ids(1, j);
59          if ~isempty(node.xy(:, cur_nd_id == nd_ids))
60              nd_coor(:, j) = node.xy(:, cur_nd_id == nd_ids);
61          end
62      end
63      % remove zeros
64      nd_coor(any(nd_coor==0,2),:)=[];
65      if ~isempty(nd_coor)
66          %plot way (highway = blue, other = green)
67          if flag == 1
68              z1=plot(hax, nd_coor(1,:), nd_coor(2,:), '-','LineWidth'
                    ,1.25,'color',[185/255 122/255 87/255]);% Plot Street
69              streets{1,zj}=[nd_coor(1,:); nd_coor(2,:)];
70              zj=zj+1;
71          else
72                          z2=plot(hax, nd_coor(1,:), nd_coor(2,:), '-'
                                ,'LineWidth',1,'color',[0/255 162/255
                                232/255]);%Recreational areas
73          end
74          if flag == 3
75              z3=plot(hax, nd_coor(1,:), nd_coor(2,:), '-','LineWidth'
                    ,1,'color',[128/255 07/255 255/255]);%Plot Houses
76                           z4=plot(hax, nd_coor(1,end), nd_coor(2,end)
                                , '<','markersize',4,'color',[1 0 0]);
77              house(zi,:)=[nd_coor(1,end) nd_coor(2,end)];
78              House{1,zi}=[nd_coor(1,:); nd_coor(2,:)];
79              zi=zi+1;
80          end
81           if flag == 4
82              plot(hax, nd_coor(1,:), nd_coor(2,:), '-','LineWidth',1,
                    'color',[1 0 0]);
83              streets2{1,zk}=[nd_coor(1,:); nd_coor(2,:)];
84              zk=zk+1;
85           end
86      end
87      %waitforbuttonpress
88  end
89  disp(key_catalog.')
90
91  function [] = disp_info(bounds, Nnode, Nway)
92  disp( ['Bounds: xmin = ' num2str(bounds(1,1)),...
93      ', xmax = ', num2str(bounds(1,2)),...
94      ', ymin = ', num2str(bounds(2,1)),...
95      ', ymax = ', num2str(bounds(2,2)) ] )
96  disp( ['Number of nodes: ' num2str(Nnode)] )
97  disp( ['Number of ways: ' num2str(Nway)] )
```

The presented Matlab algorithm named delete_data uses an iterative approach to remove unwanted data from a geospatial dataset. First, boundaries are defined for longitude and latitude, which are assigned to two variables called "lonlim" and "latlim," respectively. Next, a list of intersection vectors "int" containing information about intersections between streets in a city or geographic area is traversed. At each iteration, the x and y coordinates of the street intersections are extracted and filtered to include only those within the previously defined longitude and latitude limits. The filtered data is stored in a "deleting" matrix to store the deleted data.

In summary, the algorithm uses an iterative filtering approach to remove unwanted data from a geospatial dataset based on the boundaries defined for longitude and latitude. The algorithm runs in a loop for each intersection vector "int," extracting and filtering street intersections' x and y coordinates. The filtered data is stored in a "deleting" array for further processing or deletion. This filtering approach can help remove noisy or irrelevant data in urban geography or visualize maps of specific geographic areas. Table 2.5 presents the variables used in the algorithm 2 that eliminates data outside the analysis area; the Matlab code is shown below.

Table 2.5: Variables related to the Function delete_data

Variable	**Description**
long	A vector of longitudes
lat	A vector of latitudes
int	A cell array of street coordinates
jg	Index variable for iterating through *int*
streetsx	A vector of street longitudes
streetsy	A vector of street latitudes
tot	A matrix of selected street coordinates
ui	Index variable for iterating through *streetsx* or *tot*
tot2	A matrix of selected street coordinates after both latitude and longitude filters
deleting	A cell array of deleted street coordinates

Algorithm 2 Function delete_data

function DELETE_DATA($long, lat, int$)
  $deleting \leftarrow []$
  for $jg \leftarrow 1$ to $length(int)$ **do**
    $streetsx \leftarrow []; streetsy \leftarrow [];$
    if $length(cat(2, intjg, 1)) \neq 0$ **then**
      $streetsx, streetsy \leftarrow cat(2, intjg, 1)(1, :), cat(2, intjg, 1)(2, :)$
    end if
    $tot \leftarrow []$
    for $ui \leftarrow 1$ to $length(streetsx)$ **do**
      if $lonlim(1, 1) < streetsx(ui) < lonlim(1, 2)$ **then**
        $tot \leftarrow [tot; streetsx(ui), streetsy(ui)]$
      end if
    end for
    $tot2 \leftarrow []$
    for $ui \leftarrow 1$ to $length(tot)$ **do**
      if $latlim(1, 1) < tot(ui, 2) < latlim(1, 2)$ **then**
        $tot2 \leftarrow [tot2; tot(ui, :)]$
      end if
    end for
    $deleting\{jg, 1\} \leftarrow [tot2(:, 1)'; tot2(:, 2)']$
  end for
  return *deleting*
end function

```
function [deleting]=delete_data(long,lat,int)
lonlim=[]; latlim=[];
lonlim=long; latlim=lat;
for jg=1:length(int)
    Xtr=[]; Ytr=[]; xeb=[]; yeb=[]; tx=[]; ty=[];
    streetsx=[]; streetsy=[]; delete=[];
    if length(cat(2,int{jg,1}))~=0
        streets_1=cat(2,int{jg,1});
        streetsx=streets_1(1,:);
        streetsy=streets_1(2,:);
    end
    Xtr=streetsx; Ytr=streetsy;
    xeb=Xtr'; yeb=Ytr';
    for ui=1:length(xeb)
        filtro1=xeb(ui);
        if filtro1>lonlim(1,1) && filtro1<lonlim(1,2)
            tx(ui)=filtro1;
        else
            tx(ui)=0;
        end
    end
    tot=[tx' yeb];
    if length(tot)~=0
        delete=find(tot(:,1)==0);
        tot(delete,:)=[];
        xeb=tot(:,1);
```

```
27          yeb=tot(:,2);
28      else
29          xeb=[];
30          yeb=[];
31      end
32      ty=zeros(1,length(yeb));
33      for ui=1:length(yeb)
34          filtro2=yeb(ui);
35          if filtro2>latlim(1,1) && filtro2<latlim(1,2)
36              ty(ui)=filtro2;
37          else
38              ty(ui)=0;
39          end
40      end
41      tot2=[xeb ty'];
42      if length(tot2)~=0
43          delete2=find(tot2(:,2)==0);
44          tot2(delete2,:)=[];
45          xtr=tot2(:,1);
46          ytr=tot2(:,2);
47      else
48          xtr=[];
49          ytr=[];
50      end
51      xtr=xtr'; %stores the intersections within the scenario
52      ytr=ytr'; %stores the intersections within the scenario
53      deleting{jg,1}=[xtr;ytr];
54  end
55  end
```

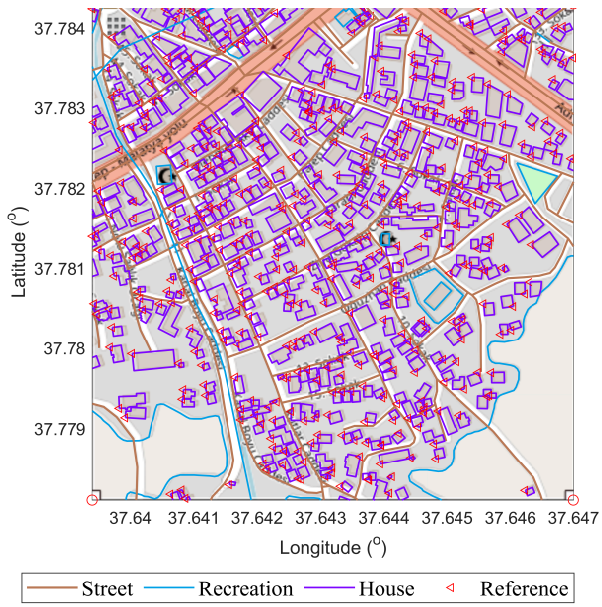


Figure 2.9: New scenario considering the location of housing, streets, and recreation areas.

Figure 2.9 represents the dwellings recovered from the OSM archive in purple. Each sector or stratum can be represented by color. The representation identifies a single color for didactic purposes. A correct planning process considers geo-referenced scenarios to determine the solution to a specific problem.

After evaluating the interface between OpenStreetMap and Matlab, we will load the information generated in the above file. For storing all the variables, you can type in the command window after obtaining the graphs the command: **save store.mat esc**. Then in a new script, we make an application to perform the routing of a set of sensors located in each house. Wireless communication is established with a maximum distance that must be met, in this case, 45 meters.

The algorithm in question uses MATLAB programming language to perform a series of operations on a data set stored in a file with a .mat extension. First, all variables in the current workspace are deleted, the screen is cleared, and all system warnings are disabled. Next, the store.mat file containing the data to be analyzed is loaded.

Subsequently, a figure is generated in which the data of the house variable is graphically represented, using the k-means algorithm to divide the data into three groups or clusters. Other symbols and colors represent the different clusters, and a reference point (centroids) is added for each cluster.

Then, a series of network analysis operations are performed using Dijkstra's technique to calculate the shortest path between all the network nodes. The distance between the different nodes is calculated using Haversine's formula, and a maximum length of 0.045 (45 meters) is set to establish the connections between the network nodes. Finally, the connections between the different nodes of the network are graphically represented, and the shortest path between each is calculated.

The presented MATLAB algorithm uses data and graph analysis techniques to graphically represent a data set and calculate the shortest path between its different nodes.

The generated between line 20 and line 40 of the code aims to calculate the Haversine distance between each pair of geographic coordinate points (latitude and longitude) provided in two different vectors. The result is stored in a distance matrix called distH, where each entry represents the distance between a pair of points. Then, a maximum distance limit (dmax) is set, and a line is drawn between two points only if their distance is less than or equal to dmax. It is done to visualize issues that are close enough to each other on a map, represented by points and lines so that they can be connected in a network graph.

In the Matlab algorithm, graph theory calculates the distance between nodes in a graph G=(V, E), where V is the set of vertices and E is the edges. The algorithm

from line 42 uses a distance matrix distH to store the distances between each pair of nodes in the graph. The line of code distH(distH==0)=inf; states that if the distance between two nodes is zero, it should be considered infinite, which prevents it from being used in path planning. Then, the line G(distH<=dmax)=1; establishes a network connectivity matrix, where nodes at a distance less than or equal to dmax are connected. Finally, Dijkstra's algorithm is used by calling the function dijkstra_A, to find the shortest path between an initial node N and all other nodes in the network G, generating a distance matrix dp and a predecessor matrix that can be used to reconstruct the shortest path between any pair of nodes. Table 2.6 presents the variables used in the algorithm's 3 pseudocode for planning a wireless sensor network in a geo-referenced scenario; the Matlab code is presented below.

Table 2.6: Variables related to the Main algorithm

Variable	**Description**
X	Input data for k-means clustering
k	Number of clusters
$idx2$	Cluster indices for each data point
C	Cluster centroids
G	Adjacency matrix for graph
n	Number of data points
$distH$	Pairwise distances between data points
lon	Longitudes for data points
lat	Latitudes for data points
$dmax$	Maximum distance threshold for connecting graph edges
$path$	Array to store paths for each node
dp	Array to store shortest distances for each node
$pred$	Array to store predecessor nodes for each node in the shortest path

Algorithm 3 Main Algorithm of Network Planning

Load store.mat
$X \leftarrow$ house
$k \leftarrow 3$
$[idx2, C] \leftarrow \text{kmeans}(X, k)$
$G \leftarrow \textbf{zeros}(\textbf{length}(\text{house}))$
$n \leftarrow \textbf{length}(X)$
$distH \leftarrow \textbf{zeros}(n, n)$
$G \leftarrow \textbf{zeros}(n, n)$
$lon \leftarrow X(:, 1)$
$lat \leftarrow X(:, 2)$
for $i \leftarrow 1$ to $n - 1$ **do**
    for $j \leftarrow i + 1$ to n **do**
        $distH(i, j) \leftarrow \text{haversine}([lat(i), lon(i)], [lat(j), lon(j)])$
        if $\sim$ **isreal**$(distH(i, j))$ **then**
            $i, j,$ **pause**
        end if
        $distH(j, i) \leftarrow distH(i, j)$
    end for
end for
$distH(distH == 0) \leftarrow \infty$
$G(distH \leq dmax) \leftarrow 1$
$path \leftarrow []$
$[dp, pred] \leftarrow \text{dijkstra\_A}(G, N)$
for $i \leftarrow 1$ to N **do**
    $node \leftarrow i$
    $pathnode \leftarrow [node]$
    $totalCost \leftarrow 0$
    while $pred(node) < N + 1$ **and** $pred(node) > 0$ **do**
        $pred(node)$
        $pathi \leftarrow [pathi, pred(node)]$
        $totalCost \leftarrow totalCost + distH(node, pred(node))$
        $node \leftarrow pred(node)$
    end while
end for

```
clc; clear all; close all;
warning('off','all');
load store.mat
%=======================================
figure(2);hold on; grid on; box ('on');
X=house;
k=3;colores=lines(k);
[idx2,C] = kmeans(X,k);
z2=plot(X(idx2==1,1),X(idx2==1,2),'h','color',colores(1,:),'
    MarkerSize',12);
plot(X(idx2==1,1),X(idx2==1,2),'.r','MarkerSize',12);
z3=plot(X(idx2==2,1),X(idx2==2,2),'o','color',colores(2,:),'
    MarkerSize',12);
plot(X(idx2==2,1),X(idx2==2,2),'.k','MarkerSize',12);
z4=plot(X(idx2==3,1),X(idx2==3,2),'s','color',colores(3,:),'
    MarkerSize',12);
plot(X(idx2==3,1),X(idx2==3,2),'.b','MarkerSize',12);
z5=plot(C(:,1),C(:,2),'kx','MarkerSize',15,'LineWidth',2);
legend('Cluster 1','Cluster 2','Centroids','Location','NW');
G=zeros(length(house));
n=length(X); distH=zeros(n,n); G=0*distH;
lon=X(:,1);lat=X(:,2);
    for i=1:n-1
        for j=(i+1):n
            distH(i,j)=haversine([lat(i) lon(i)],[lat(j) lon(j)]);
            if ~isreal(distH(i,j))
                i,j,pause;
            end
            distH(j,i)=distH(i,j);
        end
    end
%=======================================
    dmax=0.045;
    M=length(lon);
    N=length(lat);
    for i=1:N
        for j=1:N
            if distH(i,j)<=dmax
                z6=plot([lon(j) lon(i)],[lat(j) lat(i)],'-','color'
                    ,[127/255 127/255 127/255]);hold on;
            end
        end
    end
%=======================================
% Graph Theory
distH(distH==0)=inf;
G(distH<=dmax)=1;
path=[];
[dp,pred]=dijkstra_A(G,N);
for i=1:N
    node=i;
    path{node}=[node];
```

```
49      totalCost=0;
50      while pred(node)<N+1 & pred(node)>0
51          pred(node);
52          path{i}=[path{i} pred(node)];
53          totalCost=totalCost+distH(node, pred(node));
54          if length(path{i})==2
55              col=[1 0.4 0.2];
56              ancho=1;
57              if pred(node)>0 & pred(node)<=N & pred(node)>0 & pred(
                    node)<=N,col=[1 0 1]; ancho=1.15; end % Connexion
58              z7=plot([lon(node) lon(pred(node))],[lat(node) lat(pred(
                    node))],'-','color',col,'linewidth',ancho);
59              d = findobj('Color',[1 0 1]);
60          end
61          node=pred(node);hold on; grid on;
62      end
63  end
64  %=======================================
65  legend([z2,z3,z4,z5,z6,z7(1)],'Group1','Group2','Group3','Centroide'
        ,'Mesh Feasible','Routing','fontname','times new roman','fontsize
        ',13,'location','SO','orientation','horizontal');
66  %=======================================
67  figure(2);
68  hold(ax, 'off'),box('on');
69  h=gcf;
70  set(h,'PaperPositionMode','auto');
71  set(h,'PaperType','A4');
72  set(h,'PaperOrientation','landscape');
73  set(h,'Position',[10 0 800 700]);
74  set(h, 'InvertHardcopy', 'off')
75  fig = gcf;
76  fig.Color = 'white';
77  print -dpdf -r800 figure1_18
78  figure(3);
79  imagesc(G);colormap(jet);colorbar;
80  xlabel('X-axis')
81  ylabel('Y-axis')
82  hold(ax, 'on'),grid on; box('on');
83  h=gcf;
84  set(h,'PaperPositionMode','auto');
85  set(h,'PaperType','A4');
86  set(h,'PaperOrientation','landscape');
87  set(h,'Position',[10 0 500 800]);
88  set(h, 'InvertHardcopy', 'off')
89  fig = gcf;
90  fig.Color = 'white';
91  print -dpdf -r800 figure1_19
```

The Dijkstra algorithm is a weighted, directed graph search algorithm that starts at an initial node and finds the shortest path to all other nodes. The algorithm uses a priority queue data structure to maintain a set of visited nodes and another set of unvisited nodes. The algorithm's pseudocode begins by initializing the priority queue with the starting node and its zero cost. Next, the algorithm selects the node with the lowest cost from the priority queue and marks it as visited. Then, the cost of the adjacent nodes to the visited node is updated if the current cost is greater than the sum of the cost of the visited node and the weight of the edge that connects them. This process is repeated until all nodes are visited or the final node is reached. At the end of the algorithm, the shortest path from the starting node to all other nodes in the graph is obtained. Table 2.7 presents the variables used in the algorithm's pseudocode 4; the Matlab code is presented below.

Table 2.7: Variables related to the Dijkstra's algorithm

Variable	Description
V	Set of vertices in the graph
E	Set of edges in the graph
s	Start vertex
Q	Priority queue of vertices to be processed
$dist$	Array of shortest distances from s to each vertex
$prev$	Array of previous vertices on the shortest path from s to each vertex
u	Current vertex being processed
v	Neighbor of u
$length(u, v)$	Length of the edge between vertices u and v
alt	Alternative distance from s to v via u

```
1  function [d pred]=dijkstra_A(A,u)
2  % David F. Gleich
3  % Copyright, Stanford University, 2008-2009
4  if isstruct(A),
5      rp=A.rp; ci=A.ci; ai=A.ai;
6      check=0;
7  else
8      [rp ci ai]=sparse_to_csr(A); check=1;
9  end
10 if check && any(ai)<0, error('gaimc:dijkstra', ...
11         'dijkstra''s algorithm cannot handle negative edge weights.'
               ); end
12 n=length(rp)-1;
13 d=Inf*ones(n,1); T=zeros(n,1); L=zeros(n,1);
14 pred=zeros(1,length(rp)-1);
15 n=1; T(n)=u; L(u)=n; % oops, n is now the size of the heap
16 % enter the main dijkstra loop
17 d(u) = 0;
18 while n>0
19     v=T(1); ntop=T(n); T(1)=ntop; L(ntop)=1; n=n-1; % pop the head
          off the heap
```

Algorithm 4 Dijkstra's algorithm

```
1: Q ← priority queue of vertices, initialized with start vertex s
2: dist[s] ← 0
3: prev[s] ← undefined
4: for each vertex v ∈ V \ s do
5:     dist[v] ← ∞
6:     prev[v] ← undefined
7:     add v to Q
8: end for
9: while Q is not empty do
10:    u ← vertex in Q with minimum dist[u]
11:    remove u from Q
12:    for each neighbor v of u do
13:        alt ← dist[u] + length(u, v)
14:        if alt < dist[v] then
15:            dist[v] ← alt
16:            prev[v] ← u
17:            decrease-key v in Q to dist[v]
18:        end if
19:    end for
20: end while
```

```
20      k=1; kt=ntop;                       % move element T(1) down the
            heap
21      while 1,
22          i=2*k;
23          if i>n, break; end              % end of heap
24          if i==n, it=T(i);               % only one child, so skip
25          else                            % pick the smallest child
26              lc=T(i); rc=T(i+1); it=lc;
27              if d(rc)<d(lc), i=i+1; it=rc; end % right child is
                    smaller
28          end
29          if d(kt)<d(it), break;      % at correct place, so end
30          else T(k)=it; L(it)=k; T(i)=kt; L(kt)=i; k=i; % swap
31          end
32      end                                 % end heap down
33      % for each vertex adjacent to v, relax it
34      for ei=rp(v):rp(v+1)-1              % ei is the edge index
35          w=ci(ei); ew=ai(ei);            % w is the target, ew is the
                edge weight
36          % relax edge (v,w,ew)
37          if d(w)>d(v)+ew
38              d(w)=d(v)+ew; pred(w)=v;
39              % check if w is in the heap
40              k=L(w); onlyup=0;
41              if k==0
42                  % element not in heap, only move the element up the
                        heap
43                  n=n+1; T(n)=w; L(w)=n; k=n; kt=w; onlyup=1;
44              else kt=T(k);
```

```
45              end
46              % update the heap, move the element down in the heap
47              while 1 && ~onlyup,
48                  i=2*k;
49                  if i>n, break; end          % end of heap
50                  if i==n, it=T(i);           % only one child, so
                        skip
51                  else                        % pick the smallest
                        child
52                      lc=T(i); rc=T(i+1); it=lc;
53                      if d(rc)<d(lc), i=i+1; it=rc; end % right child
                            is smaller
54                  end
55                  if d(kt)<d(it), break;      % at correct place, so
                        end
56                  else T(k)=it; L(it)=k; T(i)=kt; L(kt)=i; k=i; % swap
57                  end
58              end
59              % move the element up the heap
60              j=k; tj=T(j);
61              while j>1,                       % j==1 => element at
                    top of heap
62                  j2=floor(j/2); tj2=T(j2);    % parent element
63                  if d(tj2)<d(tj), break;      % parent is smaller, so
                        done
64                  else                         % parent is larger, so
                        swap
65                      T(j2)=tj; L(tj)=j2; T(j)=tj2; L(tj2)=j; j=j2;
66                  end
67              end
68          end
69      end
70  end
```

Josiah Renfree created the Haversine function in 2010. This function calculates the distance between two geographic locations using the Haversine formula. To use the function, the user must provide two locations in latitude and longitude, which can be in degrees, minutes, and seconds or decimal format. If the user provides the locations in degrees, minutes, and seconds format, the function converts them to decimals for calculation. The function also verifies that two sites are provided, and that the locations are valid before performing distance calculations. Once the inputs have been confirmed, the function uses the Haversine formula to calculate the distance between the two locations in kilometers and then converts this value to miles and nautical miles. In summary, a haversine function is a helpful tool for calculating the distance between two geographic locations in different units of measurement. The Matlab code for the Haversine distance is presented below.

```
1  function [km nmi mi] = haversine(loc1, loc2)
2  % Created by Josiah Renfree
```

```
% May 27, 2010
%% Check user inputs
% If two inputs are given, display error
if ~isequal(nargin, 2)
    error('User must supply two location inputs')
% If two inputs are given, handle data
else
    locs = {loc1 loc2};     % Combine inputs to make checking easier
    % Cycle through to check both inputs
    for i = 1:length(locs)
        % Check inputs and convert to decimal if needed
        if ischar(locs{i})
            % Parse lat and long info from current input
            temp = regexp(locs{i}, ',', 'split');
            lat = temp{1}; lon = temp{2};
            clear temp
            locs{i} = [];           % Remove string to make room for
                array
            % Obtain degrees, minutes, seconds, and hemisphere
            temp = regexp(lat, '(\d+)\D+(\d+)\D+(\d+)(\w?)', 'tokens
                ');
            temp = temp{1};
            % Calculate latitude in decimal degrees
            locs{i}(1) = str2double(temp{1}) + str2double(temp{2})
                /60 + ...
                 str2double(temp{3})/3600;
            % Make sure hemisphere was given
            if isempty(temp{4})
                 error('No hemisphere given')
            % If latitude is south, make decimal negative
            elseif strcmpi(temp{4}, 'S')
                 locs{i}(1) = -locs{i}(1);
            end
            clear temp
            % Obtain degrees, minutes, seconds, and hemisphere
            temp = regexp(lon, '(\d+)\D+(\d+)\D+(\d+)(\w?)', 'tokens
                ');
            temp = temp{1};
            % Calculate longitude in decimal degrees
            locs{i}(2) = str2double(temp{1}) + str2double(temp{2})
                /60 + ...
                 str2double(temp{3})/3600;
            % Make sure hemisphere was given
            if isempty(temp{4})
                 error('No hemisphere given')
            % If longitude is west, make decimal negative
            elseif strcmpi(temp{4}, 'W')
                 locs{i}(2) = -locs{i}(2);
            end
            clear temp lat lon
        end
    end
```

```
50 end
51 % Check that both cells are a 2-valued array
52 if any(cellfun(@(x) ~isequal(length(x),2), locs))
53     error('Incorrect number of input coordinates')
54 end
55 % Convert all decimal degrees to radians
56 locs = cellfun(@(x) x .* pi./180, locs, 'UniformOutput', 0);
57 %% Begin calculation
58 R = 6371;                                   % Earth's radius in km
59 delta_lat = locs{2}(1) - locs{1}(1);        % difference in latitude
60 delta_lon = locs{2}(2) - locs{1}(2);        % difference in
       longitude
61 a = sin(delta_lat/2)^2 + cos(locs{1}(1)) * cos(locs{2}(1)) * ...
62     sin(delta_lon/2)^2;
63 c = 2 * atan2(sqrt(a), sqrt(1-a));
64 km = R * c;                                 % distance in km
65 %% Convert result to nautical miles and miles
66 nmi = km * 0.539956803;                     % nautical miles
67 mi = km * 0.621371192;                      % miles
```

Figure 2.10 shows the feasible mesh obtained in gray due to the maximum distance restriction allowed, in this case, 45 meters. The pink color indicates the minimum spanning tree. There are unconnected nodes, which gives rise to the Steiner Tree Problem, and Steiner nodes could allow the connectivity of all nodes. The Steiner nodes will be a set of feasible points active only if they are required to connect to the wireless sensors proposed in this scenario. The scalability in the growing populations also facilitates the sensors to be interconnected due to their proximity.

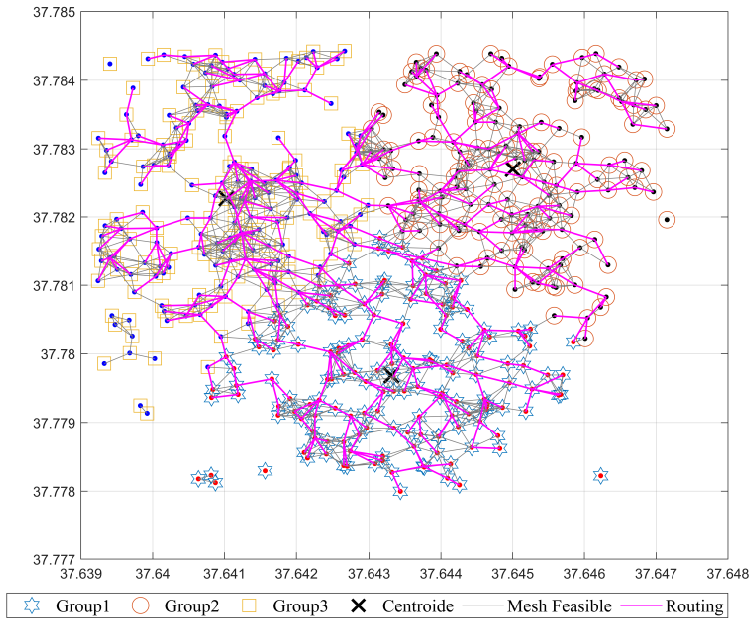


Figure 2.10: Node routing through the use of Dijkstra's Algorithm.

The connectivity matrix shown in figure 2.11 shows symmetry. Only the values in yellow have been considered possible connections in matrix G. Those represented in green are the unlabeled connections between the nodes.

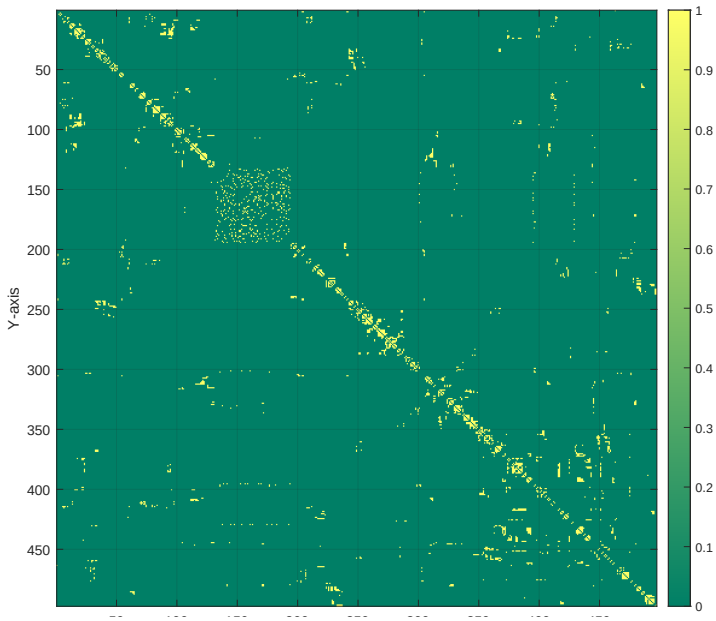


Figure 2.11: Connectivity matrix between 479 nodes.

2.5 Conclusions

This scientific article presented a method for OSM file reading in Matlab using a combination of existing tools. The limitation of direct OSM file reading by Matlab was overcome, and a structured file was obtained, allowing for efficient working with the data.

Furthermore, additional Matlab functions not found in MathWorks were explored. Several valuable functions were found that improved the efficiency of the data reading and filtering process, allowing for processing large datasets in a reasonable amount of time.

New Matlab scripts with the filtered house and dwelling data from OSM files have been generated. These scripts allow for extracting relevant information about the location and features of homes and dwellings, which is helpful in network planning.

It was demonstrated that applications for network planning in geo-referenced scenarios could be created using the data obtained from OSM files. It enables the identification of critical areas and planning for effective solutions in emergencies.

Finally, a didactic process was proposed that allows new research innovation from the information in OSM files and the generated Matlab codes. This process

guides users in identifying new research opportunities using geo-referenced data obtained from OSM files and Matlab tools. Overall, the results demonstrate the usefulness of combining free software tools and Matlab for geospatial data processing.

Georeferencing has been a valuable tool in planning various activities and has allowed complex problems to be solved efficiently. The code presented in this paper has demonstrated how georeferencing can be combined with other techniques, such as graph theory, to solve problems in different areas. In particular, it has been presented how georeferencing and graph theory can be used to solve routing and route planning problems in geographic regions, which can have practical applications in logistics, transportation, and other activities.

Notably, the code presented in this paper demonstrates the innovation of using georeferencing and provides a practical and accessible tool for solving real-world problems. In addition, the code is presented clearly and concisely, which makes it easily reproducible and adaptable to solve other similar problems in different areas. In summary, the code shown in this article is a valuable contribution to the scientific community. It can be used as a basis for future georeferencing and geographic planning work.

Bibliography

[1] L. Machado and E. Inga, "Optimal Placement of UDAP in Advanced Metering Infrastructure for Smart Metering of Electrical Energy Based on Graph Theory", *Electronics (Switzerland)*, vol. 11, no. 11, 2022, ISSN: 20799292. DOI: 10.3390/electronics11111767.

[2] E. Inga, R. Hincapié, and S. Céspedes, "Capacitated Multicommodity Flow Problem for Heterogeneous Smart Electricity Metering Communications Using Column Generation", *Energies*, vol. 13, no. 1, p. 97, 2019. DOI: 10.3390/en13010097.

[3] E. Inga, S. Céspedes, R. Hincapié, and A. Cárdenas, "Scalable Route Map for Advanced Metering Infrastructure Based on Optimal Routing of Wireless Heterogeneous Networks", *IEEE Wireless Communications*, vol. 24, no. April, pp. 1–8, 2017, ISSN: 1536-1284. DOI: 10.1109/MWC.2017.1600255. [Online]. Available: https://ieeexplore.ieee.org/document/7909154/.

[4] E. Inga, J. Inga, and A. Ortega, "Novel approach sizing and routing of wireless sensor networks for applications in smart cities", *Sensors*, vol. 21, no. 14, pp. 1–17, 2021, ISSN: 14248220. DOI: 10.3390/s21144692.

[5] E. Quintana and E. Inga, "Optimal Reconfiguration of Electrical Distribution System Using Heuristic Methods with Geopositioning Constraints", *Energies*, vol. 15, no. 15, pp. 1–20, 2022, ISSN: 19961073. DOI: 10.3390/en15155317.

[6] J. García and E. Inga, "Georeferenced rural distribution network model considering scalable growth of users in rural areas", *Heliyon*, vol. 9, no. 1, e12724, 2023, ISSN: 24058440. DOI: 10.1016/j.heliyon.2022.e12724. [Online]. Available: https://doi.org/10.1016/j.heliyon.2022.e12724.

[7] A. Valenzuela, E. Inga, and S. Simani, "Planning of a resilient underground distribution network using georeferenced data", *Energies*, vol. 12, no. 4, 2019, ISSN: 19961073. DOI: 10.3390/en12040644.

[8] H. Lara and E. Inga, "Efficient Strategies for Scalable Electrical Distribution Network Planning Considering Geopositioning", *Electronics (Switzerland)*, vol. 11, no. 19, pp. 1–15, 2022, ISSN: 20799292. DOI: 10.3390/electronics11193096.

[9] F. Pabón, E. Inga, and M. Campaña, "Planning Underground Power Distribution Networks to Minimize Negative Visual Impact in Resilient Smart Cities", *Electricity*, vol. 3, no. 3, pp. 463–479, 2022. DOI: 10.3390/electricity3030024.

[10] M. Campaña, E. Inga, and J. Cárdenas, "Optimal sizing of electric vehicle charging stations considering urban traffic flow for smart cities", *Energies*, vol. 14, no. 16, pp. 1–16, 2021, ISSN: 19961073. DOI: 10.3390/en14164933.

[11] L. Andrei and O. Luca, "Open tools for analysis of elements related to public transport performance. Case study: Tram network in Bucharest", *Applied Sciences (Switzerland)*, vol. 11, no. 21, 2021, ISSN: 20763417. DOI: 10.3390/app112110346.

[12] C. Ludwig, R. Hecht, S. Lautenbach, M. Schorcht, and A. Zipf, "Mapping public urban green spaces based on openstreetmap and sentinel-2 imagery using belief functions", *ISPRS International Journal of Geo-Information*, vol. 10, no. 4, 2021, ISSN: 22209964. DOI: 10.3390/ijgi10040251.

[13] C. Klonner, M. Hartmann, R. Dischl, *et al.*, "The sketch map tool facilitates the assessment of openstreetmap data for participatory mapping", *ISPRS International Journal of Geo-Information*, vol. 10, no. 3, 2021, ISSN: 22209964. DOI: 10.3390/ijgi10030130.

[14] R. Gaugl, S. Wogrin, U. Bachhiesl, and L. Frauenlob, "GridTool: An open-source tool to convert electricity grid data", *SoftwareX*, vol. 21, p. 101 314, 2023, ISSN: 23527110. DOI: 10.1016/j.softx.2023.101314. [Online]. Available: https://doi.org/10.1016/j.softx.2023.101314.

[15] A. N. Wu and F. Biljecki, "InstantCITY: Synthesising morphologically accurate geospatial data for urban form analysis, transfer, and quality control", *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 195, no. November 2022, pp. 90–104, 2023, ISSN: 09242716. DOI: 10.1016/j.isprsjprs.2022.11.005. [Online]. Available: https://doi.org/10.1016/j.isprsjprs.2022.11.005.

[16] J. Kim, S. Rapuri, E. Chuluunbaatar, *et al.*, "Developing and evaluating transit-based healthcare accessibility in a low- and middle-income country: A case study in Ulaanbaatar, Mongolia", *Habitat International*, vol. 131, no. December 2022, 2023, ISSN: 01973975. DOI: 10.1016/j.habitatint.2022.102729.

[17] M. I. Kersapati, "Land use-based stakeholders mapping for natural heritage preservation – Case study: Hampstead Heath Ponds, London", *Urban Forestry and Urban Greening*, vol. 80, p. 127 821, 2023, ISSN: 16108167. DOI: 10.1016/j.ufug.2022.127821. [Online]. Available: https://doi.org/10.1016/j.ufug.2022.127821.

[18] L. Amaya, "Optimal Design of Electrical Distribution Networks Using Optimization Models . Diseño Óptimo de Redes Eléctricas de Distribución Mediante Modelos de Optimización .", *Ingeniería y Competitividad*, 2023. DOI: 10.25100/iyc.v25i1.11572.

[19] S. Zourlidou, M. Sester, and S. Hu, "Recognition of Intersection Traffic Regulations from Crowdsourced Data", *ISPRS International Journal of Geo-Information*, vol. 12, no. 1, p. 4, 2022, ISSN: 22209964. DOI: 10.3390/ijgi12010004.

[20] J. Hellekes, A. Kehlbacher, M. L. Díaz, *et al.*, "Parking space inventory from above: Detection on aerial images and estimation for unobserved regions", *IET Intelligent Transport Systems*, no. November, pp. 1–13, 2022, ISSN: 17519578. DOI: 10.1049/itr2.12322.

[21] C. M. Albrecht, R. Zhang, X. Cui, *et al.*, "Change Detection from Remote Sensing to Guide OpenStreetMap Labeling", *ISPRS International Journal of Geo-Information*, vol. 9, no. 7, 2020, ISSN: 22209964. DOI: 10.3390/ijgi9070427.

[22] M. Hacar, "Analyzing the Behaviors of OpenStreetMap Volunteers in Mapping Building Polygons Using a Machine Learning Approach", *ISPRS International Journal of Geo-Information*, vol. 11, no. 1, 2022, ISSN: 22209964. DOI: 10.3390/ijgi11010070.

[23] H. Song, L. Yang, and J. Jung, "Self-Filtered Learning for Semantic Segmentation of Buildings in Remote Sensing Imagery With Noisy Labels", *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 16, pp. 1113–1129, 2022, ISSN: 21511535. DOI: 10.1109/JSTARS.2022.3230625.

[24] K. Milleville, S. Verstockt, and N. Van de Weghe, "Automatic Georeferencing of Topographic Raster Maps", *ISPRS International Journal of Geo-Information*, vol. 11, no. 7, pp. 1–17, 2022, ISSN: 22209964. DOI: 10.3390/ijgi11070387.

[25] J. Kim and E. Atkins, "Airspace Geofencing and Flight Planning for Low-Altitude, Urban, Small Unmanned Aircraft Systems", *Applied Sciences (Switzerland)*, vol. 12, no. 2, 2022, ISSN: 20763417. DOI: 10.3390/app12020576.

3. Multicriteria Analysis for Quality and Reliability in Electrical Systems

Ph.D. Alexander Aguila Téllez is a Professor of Electrical Engineering Career and the Master's Program in Electricity at the Universidad Politécnica Salesiana (Ecuador)
Email: aaguila@ups.edu.ec
https://orcid.org/0000-0001-7749-5644
DOI: 10.17163/abyaups.44.348

3.1 Introduction

Electrical Power Systems (EPS) generate, transmit and distribute electrical energy in strict operating ranges. To successfully fulfill these operational phases of the EPS, efficiency, safety, quality, and reliability standards must be established. Each of these standards is associated with variables that have to function within regulated limits. Various authors have proposed many methodologies to optimize these variables to work correctly within the established boundaries. However, most electrical variables present a relationship that can cause unwanted conflicts of interest. It is why more optimization methodologies have recently been proposed that involve the joint analysis of multiple variables (criteria) (1).

The Efficiency and Power Quality areas are closely related because most of the proposed improvement actions are associated with reducing the current flowing through the lines and reducing the line impedance's ohmic values. Regarding reliability, the proposed improvement actions are intended to reduce the number of failures per year and the restoration times of electrical service (2).

The multicriteria method proposes a robust and novel solution to problems with multiple interrelated objectives (decision criteria). In most engineering problems, mono-objective solutions affect other variables of the problem that have yet to be considered, resulting in a conflict that is often not visible if a multi-objective analysis is not carried out. The multicriteria method has application in decision-making about solutions that can be considered efficient individually. Still, it needs a joint study focused on a more authentic problem perspective (3).

Historically, the main improvement proposal to increase the efficiency and quality of electrical power systems has been reactive power compensation through optimal location and sizing of compensators. However, reactive power compensation at any voltage level affects multiple variables, which is why the impact caused by reactive compensation on each variable must be carefully analyzed. The main variables that must be studied in reactive power compensation problems are voltage profile, power losses, power factor, and voltage total harmonic distortion (1). The optimal operation of reactive power flows in electrical power systems has become a challenging problem due to the complexity of the new scenarios that present the penetration of renewable energies and energy storage. Added to this are non-linear loads, and unbalanced loads, making current circuits require new methodologies that consider the reality of the problem. Currently, the proposed compensation is dynamic, based on power electronics, and with an automated response that causes a high injection of harmonics into the network. For these reasons, it is necessary to consider all the requirements to determine the optimal location and sizing of reactive power-compensating devices at the lowest possible cost. For this, many heuristic and metaheuristic methods have been proposed that perform a search (exploration) on the study EPS presenting multiple iterations with compensation scenarios. But, this must be done with a new multi-objective consideration that allows delimiting the conflict between variables, seeking the solution that best responds to the global interest of all the decision criteria (1).

The evaluation of reliability in electrical power systems is also analyzed from multiple indicators that respond as decision criteria in the proposed improvement actions. Electrical failures harm the consumer in two different ways; the user is affected by the frequency of electrical service interruptions and the duration of the repair of these interruptions. Electric power distribution companies aim to minimize failures' frequency and duration (4). Achieving this demands significant investments that the electric power distributors must assume to guarantee the desired reliability indicators. These indicators for level 3 of Reliability in electric power systems (distribution level) are known as Average Energy Not Supplied (AENS), System Average Interruption Frequency Index (SAIFI), System Average Interruption Duration Index (SAIDI), and Customer Average Interruption Duration Index (CAIDI)

(5), (6). Measures to reduce failure rates and interruption repair times require considerable investment in human, material, and technical resources for repair and preventive and corrective maintenance (7). Different authors have established optimization methods that seek solutions for the sectioning location and automatic reconnection configuration. And these two improvement actions also produce different results in each reliability indicator. In the case of circuit reconfiguration, it also affects power flows with its quality and efficiency variables (8). The multicriteria decision methodology will allow a general optimal solution for the optimal location of the sectioning and the optimal reconnection problem. Most of the proposed methodological solutions respond to unilateral objectives of user satisfaction. Still, often the economic indicators of interest to the electricity distribution company are not considered in this analysis. For this reason, the joint analysis of all the objectives involved in this solution is vital. For this purpose, the multicriteria decision method appears as a multi-objective optimization alternative that also allows the establishment of weights on the decision criteria, an issue that provides greater flexibility for this study.

In this chapter, the multicriteria decision methodology will be presented as an alternative to solving engineering problems that require an optimal solution. The multicriteria method allows for solving optimization problems that depend on multiple variables considering the conflict that may exist between them (1), (6). Multicriteria, multi-objective optimization, or multicriteria decision-making methods solve optimization problems involving multiple conflicting objectives or criteria. In such issues, there may not be a single solution that optimizes all requirements simultaneously. Instead, there may be a set of keys, each performing differently concerning the different criteria. Multicriteria methods can help identify and evaluate this set of solutions, allowing decision-makers to choose the best balance of the conflicting criteria. Some standard multicriteria methods include the weighted sum method, the analytical hierarchy process, and the goal programming method. The multicriteria decision proposal will be applied to solve problems of Reliability and Quality of Electric power. For these studies, power flow calculations and stochastic analysis will be performed. The method's effectiveness is demonstrated in the analyzed issues, highlighting the conflicts with single-objective research. The graphic abstract of this investigation is shown in figure 3.1.

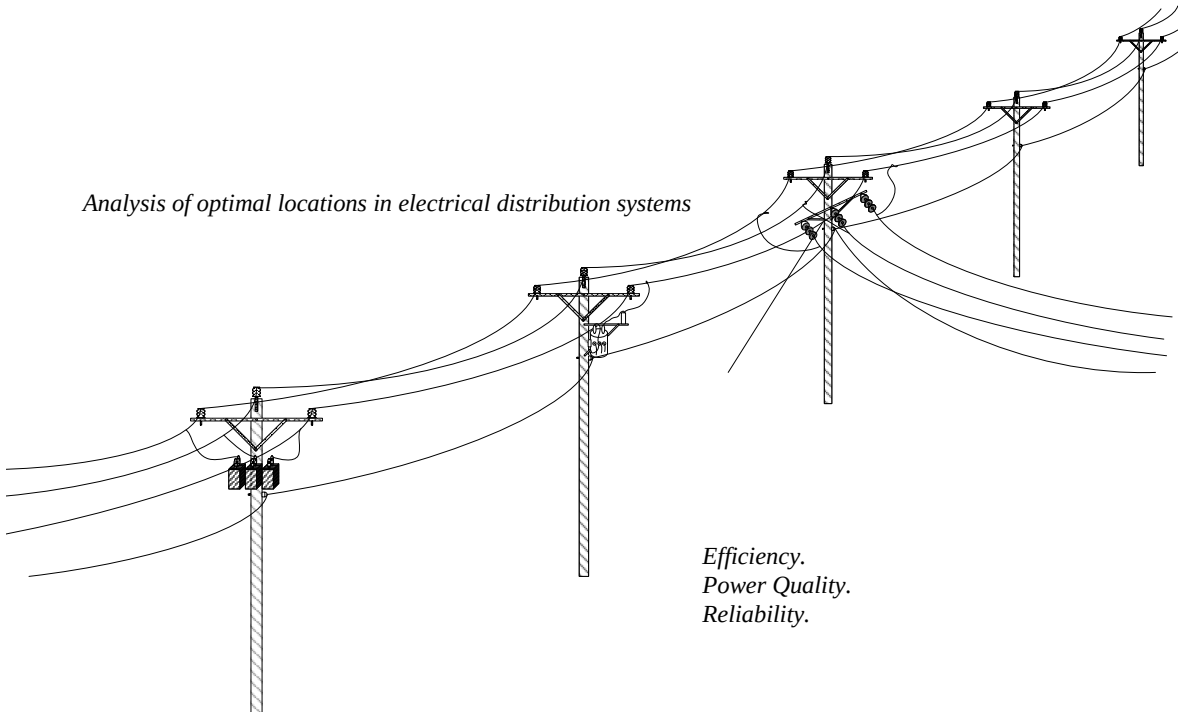


Figure 3.1: Graphical abstract of the research.

3.2 Related Works

Multicriteria decision-making (MCDM) involves evaluating and selecting alternatives based on multiple criteria or objectives, particularly relevant in electrical systems where various factors must be considered for optimal decision-making. Here's an overview of the key themes and research areas related to multicriteria decision-making in electrical systems:

Renewable Energy Integration: With the increasing penetration of renewable energy sources in power systems, researchers have focused on using multicriteria decision-making approaches to evaluate and select appropriate integration strategies. It includes the selection of optimal locations for renewable energy installations, determining suitable technologies for specific applications, and the assessment of different renewable energy scenarios.

Power System Planning and Operation: MCDM techniques have been applied to support decision-making in power system planning and operation. It evaluates alternatives for system expansion, optimal generation and transmission planning, network reconfiguration, demand response programs, and power quality improvement. Then, to make informed decisions, the goal is to consider multiple criteria

such as cost, reliability, environmental impact, and social aspects.

Smart Grid Applications: Multicriteria decision-making methods have been employed in the context of smart grids to address challenges related to advanced metering infrastructure, energy management systems, demand-side management, and grid modernization. MCDM techniques help select optimal energy efficiency solutions, demand response strategies, distributed generation integration, and load balancing.

Power Quality Assessment: MCDM approaches have been utilized to evaluate power quality issues and select appropriate mitigation strategies. The selection process considers criteria such as effectiveness, cost, and compatibility with the existing electrical system. It includes the assessment of voltage sag and swell compensation devices, harmonic filtering techniques, and reactive power compensation methods.

Fault Diagnosis and Condition Monitoring: Multicriteria decision-making methods have been applied in electrical system fault diagnosis and condition monitoring. These techniques help evaluate diagnostic options and select the most suitable method for fault detection, localization, and classification. The criteria for decision-making include accuracy, computational complexity, reliability, and cost.

Electric Vehicle (EV) Charging Infrastructure: With the rise of electric vehicles, MCDM techniques have been employed to determine optimal locations for EV charging stations, select charging technologies, and design charging infrastructure. The evaluation criteria include coverage, accessibility, charging speed, cost, and environmental impact.

These methods aim to provide a systematic and structured approach to evaluate and rank different alternatives or solutions based on their performance across multiple criteria.

Several multicriteria methods have been applied in the context of electrical power systems. Some of the commonly used methods include:

Analytic Hierarchy Process (AHP): AHP is a widely adopted multicriteria decision-making method that decomposes complex problems into a hierarchical structure and allows decision-makers to compare criteria and alternatives based on pairwise comparisons.

Technique for Order Preference by Similarity to Ideal Solution (TOPSIS): TOPSIS is a method that determines the best alternative by calculating the shortest distance to the positive ideal solution and the farthest distance to the negative perfect solution in a multidimensional space.

ELECTRE (Elimination Et Choix Traduisant la REalité): ELECTRE is a family of methods that assesses the performance of alternatives by comparing them to predefined criteria profiles. It uses pairwise comparisons and assigning concordance and

discordance indices to rank options.

PROMETHEE (Preference Ranking Organization Method for Enrichment Evaluation): PROMETHEE is a multicriteria outranking method that compares alternatives based on predefined criteria. It uses pairwise comparisons and assigns preference indices to determine the overall ranking.

Fuzzy Sets Theory: Fuzzy sets theory is often employed to handle imprecise or uncertain data in multicriteria decision-making. It allows for a more flexible representation of preferences and can take subjective judgments.

These are just a few examples of multicriteria methods applied to electrical power systems problem-solving. Researchers have explored and developed various variations and combinations of these methods to address different decision-making scenarios within the field.

The multicriteria decision technique offers several advantages over other optimization methods for decision-making in complex problems. Here are some key benefits:

Consideration of Multiple Criteria: One of the primary advantages of multicriteria decision techniques is their ability to handle multiple criteria simultaneously. Traditional optimization methods focus on a single objective, neglecting other important factors. In contrast, multicriteria decision techniques allow decision-makers to consider and weigh multiple criteria explicitly, capturing the complexity of real-world problems more comprehensively.

Trade-off Analysis: Multicriteria decision techniques facilitate trade-off analysis between different criteria. They provide a systematic framework to evaluate the performance of alternatives across multiple measures and assess the trade-offs between them. It enables decision-makers to explore different scenarios and make informed decisions that balance conflicting objectives.

Subjectivity and Stakeholder Involvement: Multicriteria decision techniques accommodate subjective preferences and incorporate stakeholder involvement in decision-making. These methods allow decision-makers to express their preferences and priorities for different criteria, ensuring that the final decision aligns with their values. Stakeholders can participate in evaluating and weighing criteria, enhancing transparency, and increasing acceptance of the final decision.

Flexibility and Adaptability: Multicriteria decision techniques offer flexibility in adapting to different problem contexts. They can handle various criteria, whether quantitative or qualitative, objective or subjective. These methods can also incorporate uncertainty and sensitivity analysis, enabling decision-makers to assess the robustness of their decisions under varying conditions.

Transparency and Communication: Multicriteria decision techniques provide a transparent and structured decision-making process. They offer clear method-

ologies for evaluating alternatives, weighting criteria, and generating rankings or preferences. This transparency enhances communication among decision-makers and stakeholders, fostering a shared understanding of the decision-making rationale and promoting consensus-building.

Holistic Decision-Making: Multicriteria decision techniques promote a holistic approach to decision-making. By considering multiple criteria, these methods consider various aspects and impacts of the decision. It leads to more comprehensive and well-rounded solutions that address the complexities and interdependencies of real-world problems.

While multicriteria decision techniques have these advantages, it's important to note that the choice of optimization method depends on the specific problem and context. Some optimization methods, such as mathematical programming or evolutionary algorithms, may be more suitable for particular optimization problems, especially those with well-defined mathematical formulations and single objectives. It is crucial to assess the characteristics of the problem and select the most appropriate method accordingly.

3.3 Problem Formulation and Methodology

Various methodologies and approaches can be employed when optimizing electrical networks using multicriteria decision-making. Here is a general outline of a solution methodology that incorporates multicriteria decision optimization for electrical networks:

Problem Formulation: Clearly define the objectives and constraints of the optimization problem. Identify the criteria that need to be considered for decision-making in electrical networks. These criteria may include cost, reliability, voltage stability, power quality, environmental impact, and system efficiency.

Model Development: Develop a mathematical model representing the electrical network and incorporating the identified criteria. This model should consider the network topology, power flow equations, and other relevant constraints. The model should also include the decision variables that can be optimized, such as generation and load dispatch, capacitor placement, and network reconfiguration.

Objective Functions: Define objective functions that quantify the performance of the electrical network concerning the identified criteria. Each criterion may have its objective function. These objective functions should reflect the desired trade-offs between the measures.

Data Collection: Gather data on the electrical network, including load profiles, generation capacities, equipment characteristics, and cost information. Ensure that the data is accurate and up-to-date to achieve reliable optimization results.

Multicriteria Decision-Making: Apply appropriate multicriteria decision-making methods to evaluate and rank different optimization alternatives. Commonly used methods include Analytic Hierarchy Process (AHP), Technique for Order Preference by Similarity to Ideal Solution (TOPSIS), ELECTRE, and PROMETHEE. These methods help to weigh the criteria, assess the performance of each alternative, and determine the optimal solution based on the decision-makers preferences.

Optimization Algorithm: Utilize optimization algorithms to find the optimal solution that satisfies the defined objectives and constraints. Different algorithms can be used depending on the complexity of the problem, such as linear programming, mixed-integer programming, or evolutionary algorithms like genetic algorithms or particle swarm optimization.

Sensitivity Analysis: Perform sensitivity analysis to examine the impact of parameter variations and uncertainties on the optimization results. This analysis helps to assess the robustness and stability of the obtained solutions.

Validation and Implementation: Validate the optimized solutions through simulations and compare the results with existing network configurations or historical data. Once validated, the answers can be implemented in the electrical network, considering any practical limitations or operational considerations.

It is important to note that the specific methodologies and tools used for multicriteria decision optimization in electrical networks may vary depending on the scope and complexity of the problem. Researchers and practitioners often tailor the method to suit their needs and requirements.

The multicriteria decision methodology is proposed for this study with the following specifications. The solution to optimization problems using the multicriteria decision technique consists of finding the best solution vector (analysis scenario) within several suitable options (whole scenarios). This vector must be determined by decision criteria established in the objective vectors. The results of each criterion for each analysis scenario can be normalized using statistical normalization techniques. In this investigation, the normalization by ranges, also known as the method of the minimum and maximum values (MM), is shown in 3.1.

$$XiNorm = \frac{Xi - Xmin}{Xmax - Xmin} \tag{3.1}$$

The CRITIC method is a problem-solving used in decision analysis and decision-making. CRITIC stands for Criteria, Ratings, and Importance. It provides a structured approach to evaluating alternatives based on multiple criteria. This method determines the relative importance or weight of each criterion. Not all criteria are equally important; this step helps you prioritize them. Assign weights to each criterion based on your judgment or through a more systematic approach. The CRITIC

method helps decision-makers organize their thinking, consider multiple criteria, and make more informed and structured decisions. It provides a systematic framework for evaluating alternatives based on performance across various dimensions or criteria (3). Once the normalized decision matrix is available, the weightings can be established using the equation 3.2.

$$Wi = Si \sum_{1}^{n} (1 - rij) \tag{3.2}$$

Where:

Wi is the weight of criterion i.

Si is the standard deviation of the data of alternatives to criterion i.

rij is the correlation coefficient between row i and column j.

Finally, the decision vector is obtained by establishing the weighted sums for each alternative; this is achieved by multiplying the result of each criterion within a scenario by the weight of that criterion. Then all scenario-weighted values for each criterion must be added; this would convert the decision matrix into a weighted decision vector. The one with the maximum or minimum value will be chosen as the winning alternative depending on whether the criteria are variables to be maximized or minimized. In the vector resulting from weighted sums (1). This calculation is shown in 3.3.

$$Pondi = \sum_{i}^{m} \sum_{j}^{n} (Wi * Xij) \tag{3.3}$$

The following figure 3.2 shows the conceptual diagram of the method to be applied.

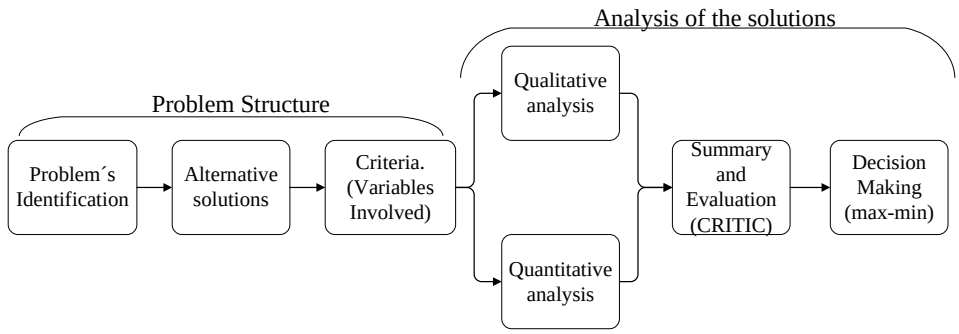


Figure 3.2: Graphical abstract of the research.

3.4 Analysis of Results

3.4.1 Optimal location and dimensioning of capacitors in microgrids using a multicriteria decision algorithm

The multicriteria optimization technique is applied to the solutions obtained for each analysis variable derived from the power flows calculated for each scenario. The algorithm assigns a specific compensation capacitance to one of the candidate nodes for analysis. The power flows are recalculated by injecting reactive power through this capacitance, yielding a result for each objective variable.

This iterative process continues until all proposed capacitances (discrete vector) have been considered for each candidate node. The Exhaustive Search algorithm (Brute Force) is justified in this analysis due to its precision, as it explores the entire set of feasible solutions. Moreover, the search space is relatively small, consisting of 14 different capacitances to be installed in 6 candidate nodes, resulting in 84 iterations. This number determines the length of the decision matrix columns. Additionally, since this is a problem related to future compensation planning, the computational time requirements are relatively simple.

The candidate nodes selected for compensation are exclusively load nodes and do not have any existing source of reactive power injection. The discrete vector of proposed capacitances to be installed is defined in steps of 50 kvar, ranging from 50 kvar to a maximum consumption of 700 kvar, representing the system's highest reactive power demand. The total cost per installation of compensating devices is shown in the equation 3.4.

$$Cost = \sum_{i=1}^{n} C\,|Qi| \geq 0 \tag{3.4}$$

Cost of installation of the reactive power, considering that C=25 USD by kvar

$$50kvar \leq |Qi| \leq 700kvar \tag{3.5}$$

Discrete vector of capacitances to be installed in steps of 50 kvar.

In addition to the problem's existing constraints, two additional considerations are made: Firstly, all nodes are considered equal candidates for connection, without any preference or variation in cost for installing the compensation devices. Secondly, the load is modeled as a constant power load, and the analysis focuses on the maximum demand scenario, ensuring that all variables meet their respective limits even in the minimum demand scenario. The variables calculated and verified for each demand scenario, in addition to the cost, are as follows:

$$Ploss = \sum_{i=1}^{n}(Pgi) - \sum_{i=1}^{n}(Pci) \geq 0 \tag{3.6}$$

Pgi is the active power generated at node i, Pci is the active power demanded in node i by each load connected, and $Ploss$ are the total losses of active power in the microgrid. Pgi, Pci and $Ploss$ are given in kW.

$$\Phi = \arctan \frac{\sum_{i=1}^{n} Qci}{\sum_{i=1}^{n} Pci} \tag{3.7}$$

Where Φ is the microgrid power factor angle measured in the bus where it is coupled to the system, Pci is the active power (in kW) demanded by each load in node i, and Qci is the reactive power (in kvar) demanded by each load in node i.

$$DPV = \frac{\sum_{i=1}^{n} |Vdi - Vi|}{n} \tag{3.8}$$

The average deviation of the voltage in the Microgrid, where n is the number of nodes of the System and Vi is the voltage in bus i in P.U. (per unit).

$$DMV = max \sum_{i=1}^{n} |Vdi - Vi| \tag{3.9}$$

The maximum deviation of the voltage, where n is the number of nodes of the System, Vi is the voltage in bus i in P.U. (per unit), and Vdi is the desired voltage in bus i in P.U.

$$THDv = \frac{\sum_{i=1}^{n} |Vi, h|}{Vi, 1} \tag{3.10}$$

Total harmonic distortion index, where Vi, h is the voltage component corresponding to harmonic h at node i, $Vi, 1$ is the fundamental component of the voltage (1st harmonic) at node i, h is the maximum harmonic order to be considered in the calculation. As constraints to the problem, it is discarded solutions that, at maximum or minimum demand, do not meet the following limits:

$$0,95 \leq Vi \geq 1,05 \tag{3.11}$$

$$THDv_max \leq 5\% \tag{3.12}$$

Therefore, it is necessary to construct a decision matrix to determine the optimal dimensioning and location of the compensating device. The decision matrix comprises n columns representing the viable alternatives for all analyzed compensation options. These alternatives adhere to the criteria of being distinct, exclusive, and exhaustive, and they define different dimensioning and locations of the compensating devices at various nodes within the system. The m rows of the decision matrix represent the quantitative criteria defined by the variables used as objective functions. Based on the established decision criteria, the optimal option is selected by initially discarding all solutions that are inferior to others using the dominance criterion implemented in the algorithm. Additionally, solutions that violate the predefined constraints previously defined as limits for these variables are eliminated. Utilizing the decision matrix facilitates the evaluation and comparison of different compensation options, allowing for the identification of the optimal solution that fulfills the established criteria and constraints.

For the application and validation of the proposed methodology, an analysis of the location and optimal sizing of reactive power compensating devices in an electrical distribution system will first be carried out.

As a base case study for this evaluation, a very novel validated 15-bar Microgrid system of which the variables involved in this analysis are known. This test system has unbalanced loads, different voltage levels, a DC bus, non-linear loads, and distributed generation. All the details about the test system chosen as a case study can be found in (9), and the resolution with the algorithms is in (10).

Then, to apply this technique effectively, it is crucial to precisely determine the decision criteria and their quantifiable scales. The alternatives should then be evaluated based on their weights for each qualitative or quantitative criterion to form the eligible set. The subsequent step involves establishing a decision matrix to select the optimal solution, encompassing the decision criteria for the entire analyzed search space. Given that the system is relatively small, an Exhaustive Search algorithm will be implemented in this case study.

After obtaining the decision matrix, the solutions will undergo dominance analysis, eliminating any solutions inferior to others in all variables. Considering the weights assigned to each variable, the weighted sums technique will be employed for the multicriteria decision-making process regarding the best location and dimensioning alternative for compensating devices.

The description of the variables used in 5 is presented in 3.1, while 3.2 outlines the variables utilized in Algorithm 2. Subsequently, the following section explains the algorithms employed to solve the multicriteria problem of reactive power compensation. 5 focuses on allocation using the exhaustive search technique, while 6 describes the multicriteria decision technique under the proposed methodology.

Table 3.1: Variables of algorithm I.

Symbol	Description
n	Bus number
$Phase$	Number of phases
$Candidate_bus$	Number of candidate buses of the HMG
$Capacitor$	Vector of capacitances
$Comp-6...Comp-n$	Selection variable of the capacitor in bus n.
$Iterations$	Number of iterations to be performed.
$VoltageLL-pu-abc$	Data of phase voltages.
$Power-Losse-Total-n$	Data of total losses for the interactions of bus n.
$DPVS-n$	Average voltage deviation
$DVmax-n$	Maximum voltage deviation

Algorithm 5 Improved brute force for the n buses

Input: $n, Candidate_bus, Capacitor$
Output: $PF, P_Losse, DPVS, DVmax, THDv, Cost$
Initialize: $Capacitor = [100e3 : 100e3 : 2000e3]; Comp_6 = 0; Comp_9 = 0;$ **until** $Comp_n = 0; Phase = 3;$
$Foralli = 1 until Candidate_bus If i = 1 then n = 6; Comp_i = 1; Comp_i + 1 until Comp_n = 0;$
$Forall j = 1 until size(Capacitor) Qc_Cap = Capacitor(j);$ $Execute the model(HMGSimulink model); Read the DATA;$
$Calculate Vpu_i, PF_i and P_Losse_i; Calculate DPVS_i and DVmax_i$
Save: $Vpu_i; PF_i; PLosse_i;$
$Endforall$
$: ElseIf i = 2 then : n = 9; Comp_i = 1; Comp_i - 1 until Comp_n = 0; Repeat steps : 5 thru 6;$
$ElseIf i = 3 then n = 10; Comp_i = 1; Comp_i - 2 until Comp_n = 0; Repeat steps : 5 thru 6;$
$ElseIf i = 4 then n = 11; Comp_i = 1; Comp_i - 3 until Comp_n = 0; Repeat steps : 5 thru 6;$
$ElseIf i = 5 then n = 12; Comp_i = 1; Comp_i - 4 until Comp_n = 0; Repeat steps : 5 thru 6;$
$ElseIf i = 6 then n = 14; Comp_i = 1; Comp_i - 5 until Comp_n - 1 = 0; Repeat steps : 5 thru 6;$ $Endif; Endforall;$
$: VoltageLL_p u_a bc_n = [VoltageLL_p u_a bc_1, VoltageLL_p u_a bc_2, ...VoltageLL_p u_a bc_i];$
$: PF_a bc_n = [PF_a bc_1, PF_a bc_2, PF_a bc_i];$
$: Power_Losse_n = [P_Losse_1, P_Losse_2, P_Losse_i];$
$: PF_a bc_n = [PF_a bc_1, PF_a bc_2, PF_a bc_i];$
Return: $V_p u_n; PF_a bc_n; PLosse_n and PF_n;$

Table 3.2: Variables of algorithm II.

Symbol	Description
$win-case$	Winning case
N	Number of observations
$DPVS$	Average deviation of system voltage
$DVmax$	Maximum deviation of system voltage
$Power-Losse-Total$	Total losses of active power
$angle$	Power factor angle
$THDv$	Total harmonic distortion of the system
$Cost$	Costs of reactive power compensation
$DPVS-Pond$	Weight of the average voltage deviation
$DVmax-Pond$	Weight of the maximum voltage deviation
$Power-Losse-Total-Pond$	Weight of the total losses in the system
$angle-Pond$	Weight of the power factor angle
$THDv-Pond$	Weight of the Total Harmonic Distortion
$Cost_Pond$	Weight of the compensation cost

Reactive power compensation has been a long-standing issue in electrical power systems. Numerous methodologies have been proposed to determine capacitor banks' optimal placement and power rating to meet quality requirements. Optimization algorithms allow for proposing the capacitor bank with the lowest reactive power demand, positioned in the most favorable location within the electric system. This challenge also extends to microgrids. The most common approach in distribution systems to compensate for reactive power is the installation of capacitor banks, which offer a simple and cost-effective solution to inject reactive power when needed. Some researchers also suggest adjusting transformer

Algorithm 6 Decision algorithm.

1: **Input:**$V_pu_n; PF_n; P_Losse_nandPF_n$;
2: **Output:**win_case;
3: **Initialize:**$Cost = [1:50:700]*25; DPVS_Pond = 0.2; DVmax_Pond = 0.4; Power_LosseTotal_Pond = 0.7; angle_Pond = 1; THDv_Pond = 1; Cost_Pond = 1.2; cont = 0;\ A = [DPVS_1, DPVS2, ..., DPVS_n - 2, DPVS_n - 1, DPVS_n;\ DVmax_1, DVmax_2, ..., DVmax_n - 2, DVmax_n - 1, DVmax_n;\ Power_LosseTotal_1, Power_LosseTotal_2, ..., Power_LosseTotal_n - 2, Power_LosseTotal_n - 1Power_LosseTotal_n;\ acos(PF_1), acos(PF_1), acos(PF_n - 2), acos(PF_n - 1), acos(PF_n);\ THDv_1, THDv_2, THDv_n - 2, THDv_n - 1, THDv_n;\ CostCostCostCostCostCost]$;
4: $Forallu = 1untildimension(coders)$
5: $if(1 - A(2,u)) < 0.95(VoltageLimitConstraint)$
6: $A(:,u) = []; (Undesiredresultsareeliminated)$
7: $Endforall; Endforall;$

capacities to measure reactive power compensation, given their high reactive power consumption as highly inductive machines.

Analogous to traditional electric systems, determining the appropriate bus for connecting a compensation bank with the least reactive power requirement is crucial to achieving desired power quality conditions. It is essential to consider both maximum and minimum demand scenarios since a candidate solution that works well in an isolated microgrid during peak demand may not be suitable during periods of low demand without connection to the main grid if it violates any power quality conditions considered in the microgrid.

As part of the exhaustive search multi-objective algorithm, a predefined set of reactive power values (candidates) and bus options (candidates) for connecting the bank are established. However, selecting a capacitor that injects a discrete amount of reactive power at a bus may sometimes be optimal. A larger or smaller reactive power value could provide better power quality. Such reactive power values may have yet to be considered among the predefined candidates. Consequently, only discrete and commercially available capacitive values are typically considered, while results with continuous capacitive values are disregarded.

There are alternative proposals for reactive power compensation in electric power systems, such as SVC (Static Var Compensator) based on thyristors, which regulate the conduction time in coils or capacitors (Thyristor Controlled Reactor, TCR, and Thyristor Switched Capacitors, TSC). Another solution involves compensation using converters that rely on static switches like the IGBT. An example is the STATCOM (Static Synchronous Compensator), which injects reactive power in a non-discretized manner.

These proposals rely on the operation of power electronics equipment. In the case of SVC, the thyristor-based process introduces additional harmonics along with the fundamental frequency, often necessitating the inclusion of filters (42), (43). The control systems of STATCOM can be more complex, making it a justifiable choice when high-speed control is a critical factor.

This research focuses on a specific case study (3) that has been chosen due to its high realism and comprehensiveness, making it an ideal subject for analysis in the context of reactive power compensation. Among numerous options, this microgrid case study provides extensive data on electrical variables, allowing for a practical and accurate analysis in scenarios where conflicts arise between the variables under consideration. The microgrid combines single-phase and three-phase loads, including non-linear and unbalanced loads. It further adds to its complexity and relevance in studying reactive power compensation. At bus 9 of the system, a non-linear load exists as a pulse-width modulated (PWM) three-phase rectifier. This rectifier operates at a commutation frequency of 4080 Hz, with

a modulation index of 0.8. It is important to note that the rectifier operates in an open-loop configuration, maintaining a fixed angle throughout its operation. The Microgrid consists of a total of 7 transformers and 14 distribution lines. Out of these transformers, 2 serve as step-down transformers, connecting the medium voltage grid to the low voltage grid. Another 3 step-down transformers facilitate the interface at low voltage with the distributed generation sources. A step-down transformer is also responsible for connecting to the Critical Consumption Point (CCP), while a step-up transformer links the Diesel generator to the medium voltage grid. Regarding distribution lines, 7 lines are operating at medium voltage, and 7 are operating at low voltage, making 14 distribution lines in the Microgrid. The one-line diagram of this case study is shown in figure 3.3.

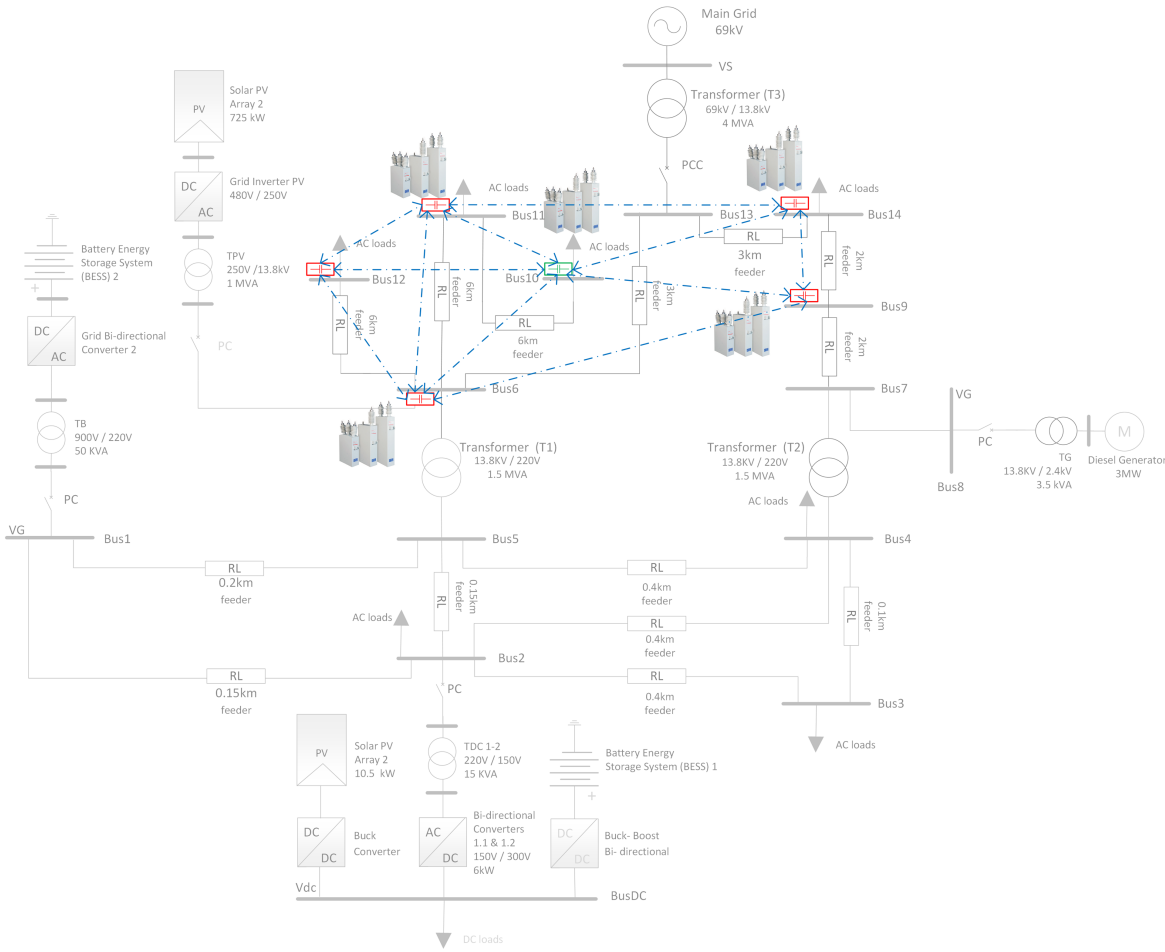


Figure 3.3: One-line diagram of the microgrid case study.

The proposed methodology, incorporating the Exhaustive Search algorithm for power flow evaluation and simulation with connected reactive capacitances and

the Multicriteria decision algorithm for determining the optimal dimensioning and location of the capacitor bank in the microgrid, has yielded groundbreaking results, which will be discussed in this section.

Figure 3.4 illustrates the normalized performance of the variables (criteria) for each of the 84 compensation scenarios, represented in a range-normalized matrix. Visually, it can be observed that similar reactive compensations yield similar behaviors across the variables, albeit differing based on the bus to which the reactive power compensation is connected. Each bus exhibits distinct outcomes.

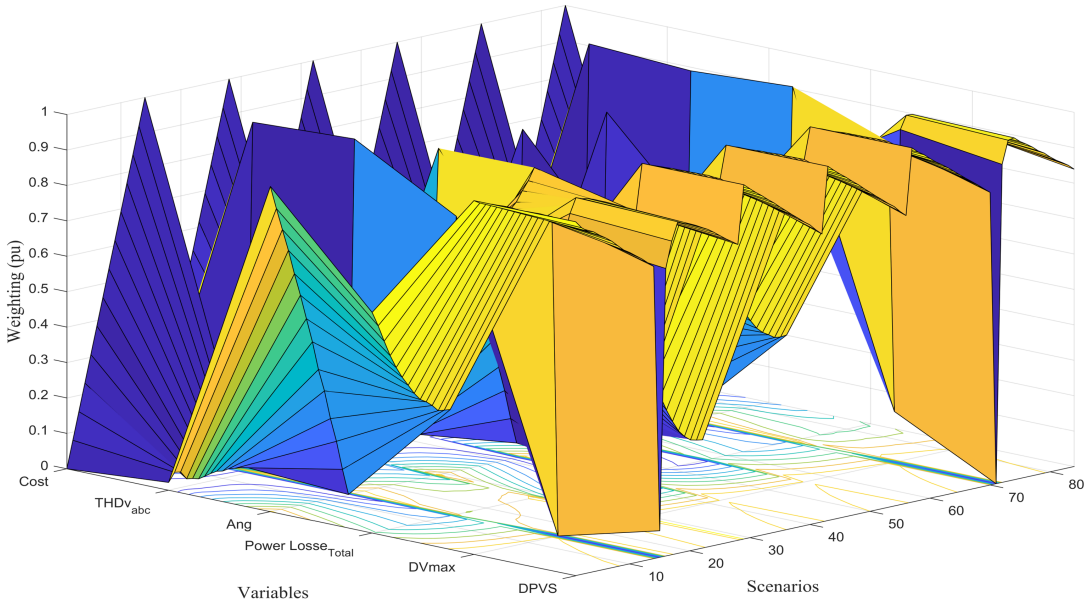


Figure 3.4: Normalized criteria for reactive power compensation scenarios.

A weighted calculation is performed per the proposed model's methodology using the normalized criteria matrix obtained for each compensation scenario. The normalized unit criteria for each variable and their corresponding weights are summed, resulting in a new vector known as the weighted sums. Figure 3.5 depicts the outcome of this vector, showcasing the weighted criteria for each compensation scenario.

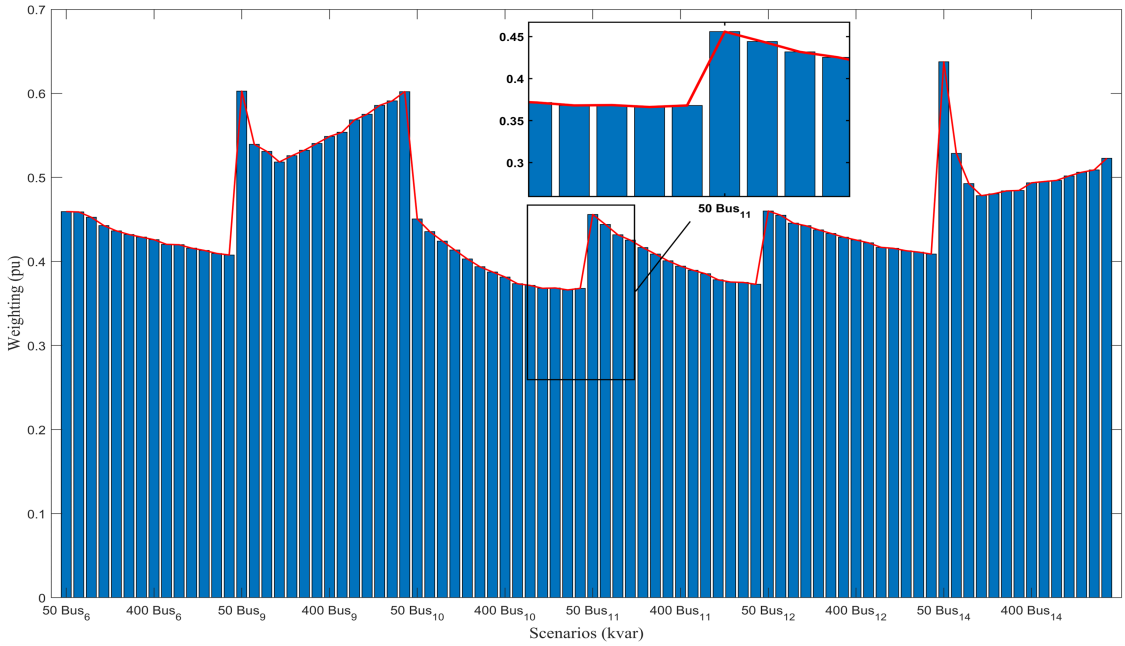


Figure 3.5: Weighted sums for each reactive power compensation scenario.

Scenario 41 emerges as the winning alternative, characterized by a reactive capacitive compensation of 650 kvar connected to bus 10 in the microgrid. The x-axis represents the capacities and locations of the capacitor. Having identified the multicriteria winning alternative (650 three-phase kvar at bus 10), each variable is analyzed to ensure compliance with operational limits (constraints) in both maximum and minimum demand load scenarios. It is worth noting that any individual results that fail to meet the constraints were eliminated during the decision algorithm, as outlined in step 4 of the algorithm.

In the analysis case where THD is not considered an objective criterion within the multicriteria optimization, Figure 3.6 illustrates the solutions represented by weighted sums for each of the 84 reactive power compensation scenarios. This analysis focuses on five variables, excluding THD, to demonstrate the inherent conflict between these variables and to highlight potential optimal outcomes that may not be truly optimal.

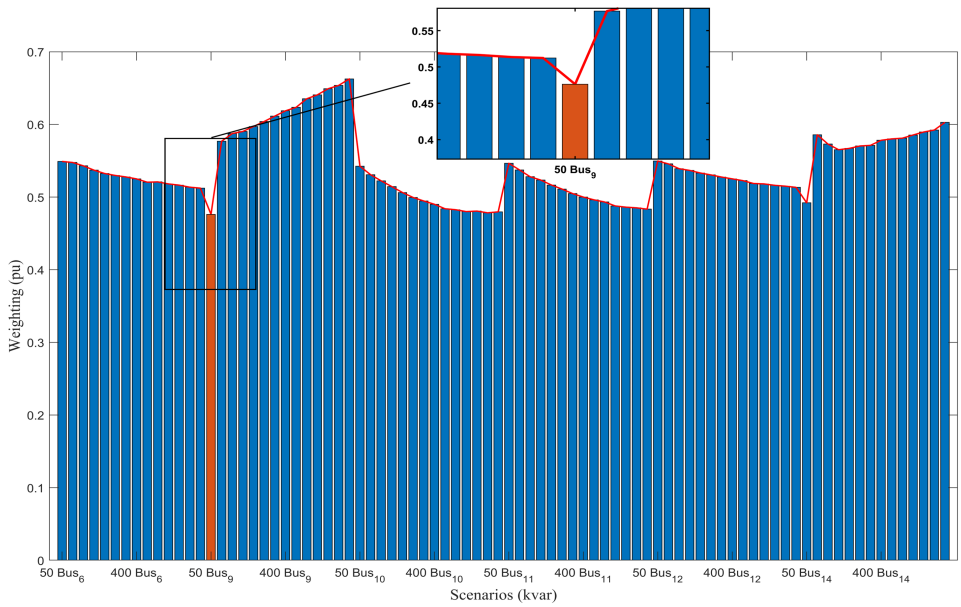


Figure 3.6: Weighted sums for each scenario of reactive power compensation without considering the THD as an objective criterion. Non-coherent result.

It is worth noting that one particular result appears "apparently" favorable and closely resembles the winning alternative previously discussed. This outcome is achieved with a minimum compensation of 50 kvar on bus 9. However, this seemingly positive result is undesired due to harmonic distortion resulting from non-linear loads at that bus. This situation creates a conflict with the THD variable. This undesired result emphasizes the problem addressed in this research, highlighting the necessity of treating reactive power compensation as a multicriteria optimization problem that accounts for variables such as harmonic distortion. It underscores the importance of considering multiple criteria to obtain truly optimal solutions.

The conflict between harmonic distortion (THD) and the Maximum Voltage Deviation variable can be observed in Figure 3.7. In this analysis, THD is excluded to independently assess each compensation scenario for bus 9. The harmonic distortion profile of the microgrid under study is compared with the Maximum Voltage Deviation variable, one of the objectives in this specific analysis comprising five objective functions.

Figure 3.7a) illustrates the THD variable for each of the 14 buses in the microgrid across the 14 compensation scenarios considered for bus 9. On the other hand, Figure 3.7b) displays the Maximum Voltage Deviation for the entire microgrid across the same 14 compensation scenarios for bus 9.

This analysis reveals a winning alternative with a minimum compensation of

50 kvar at bus 9. However, this result is undesirable due to an increase in voltage profiles caused by the harmonic distortion resulting from the connection of non-linear loads. It demonstrates the necessity of incorporating the THD variable as a weighted criterion in the multicriteria decision process. The effect of the minimum value of the maximum voltage deviation is driven by high total harmonic distortion, emphasizing the importance of considering THD in the decision-making process.

Figure 3.7: (a) THD per bus in maximum demand with compensation in bus 9 and (b) Maximum voltage deviation with compensation in bus 9 without considering the THD as an objective criterion.

Figure 3.8 illustrates the steady-state voltage waveform with 50 kvar compensation at bus 9. The voltage waveform undergoes significant distortion due to the non-linear load components present at this bus. The harmonic distortion effect on the voltage waveform led to an undesired winning alternative in the multicriteria decision process. This scenario resulted in minimum cost and minimum value of maximum voltage deviation due to the distortion-induced increase in peak voltage. The installed capacitance in this scenario had values close to the system's resonance frequency.

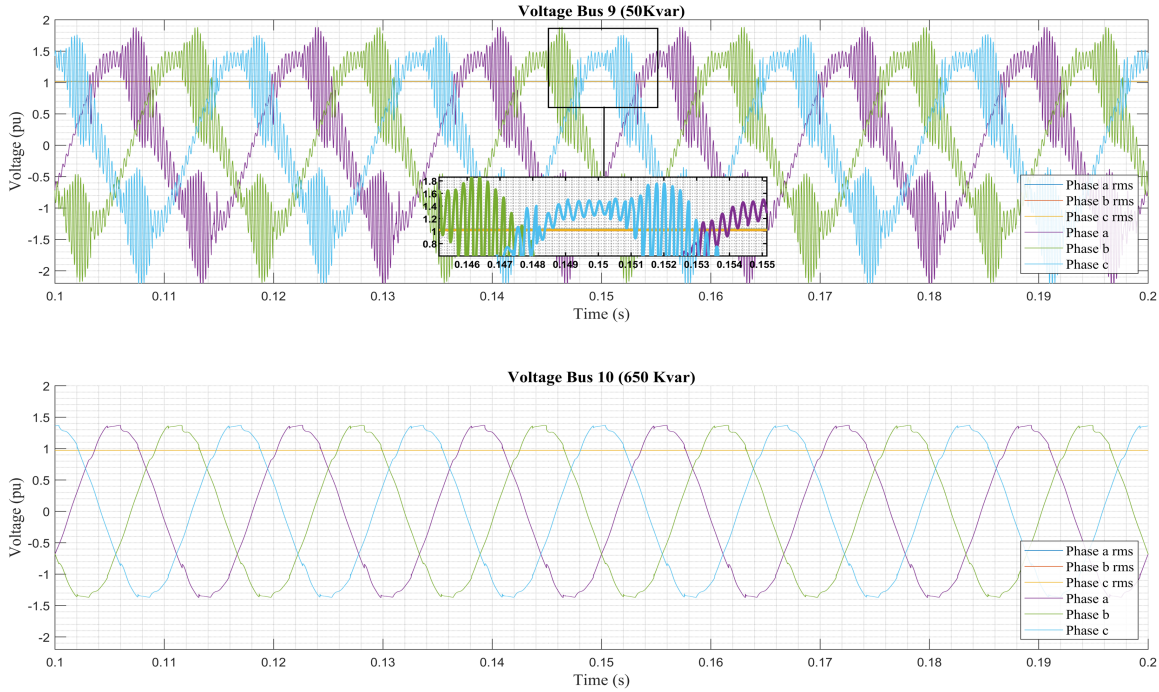


Figure 3.8: Waveforms comparison: (above) voltage at bus 9 in the 50 kvar candidate compensation and (down) voltage at bus 10 in the 650 kvar winning compensation.

This research focuses on a comprehensive case study encompassing various variables impacting a hybrid microgrid's quality and efficiency. These variables include unbalanced, non-linear, and maximum and minimum demand scenarios. The objective is to determine the optimal reactive power compensation using capacitor banks. Previous studies have approached reactive compensation as a single-objective optimization problem. However, improving the voltage profile can impact other variables contributing to an electrical system's power quality. Therefore, it is crucial to simultaneously consider variables such as voltage profile, active power losses, global power factor, and harmonic distortion rate when deciding on capacitance, dimensioning, and compensation cost.

In this work, we formulate the reactive power compensation as a multicriteria optimization problem. The algorithm considers two demand scenarios and calculates power flows with discretized compensation using the Exhaustive Search algorithm. The algorithm determines the optimal locations for discretized capacitances among candidate buses. Based on the preliminary results, a second algorithm is developed to assign weights to each variable, tailored for a minimization criterion. As a result, each compensation scenario is assigned a weighted solution, with some scenarios discarded based on the dominance criterion. The final step is to choose

the optimal solution that satisfies the defined criteria. This methodology ensures the compensation solution aligns with the correct operational requirements for reactive power compensation in distribution systems with distributed resources.

Additionally, a case is analyzed where the total harmonic distortion (THD) is not considered an objective criterion. The results highlight the conflicts between the analysis variables within the studied system. This finding justifies and supports the proposed methodology, demonstrating the importance of considering more quality and efficiency variables than previously addressed by other researchers.

Overall, the proposed methodology provides innovative solutions that underscore the need to approach the problem of reactive power compensation as a multicriteria problem, considering a comprehensive set of quality and efficiency variables.

The resolution with the algorithms is in (10).

3.4.2 Optimal location of reclosers in electrical distribution systems considering multicriteria decision

The proposed multicriteria technique will also be evaluated for a reliability problem in distribution systems (level 2).

This section introduces a novel methodology for optimizing the placement of reclosers in electric distribution systems. The proposed approach employs a multicriteria analysis to assess the reliability indicators at the distribution level (level 3) by generating scenarios using pseudo-random variables. The reliability indicators considered in this study include the System Average Interruption Frequency Index (SAIFI), System Average Interruption Duration Index (SAIDI), Customer Average Interruption Duration Index (CAIDI), and Average Energy Not Supplied (AENS).

Deterministic random generation is utilized within specified ranges associated with each analysis variable, such as failure rates of elements and loads, failure duration, number of customers per load point, mean power consumption at each load point, and others. The reliability indicators are calculated for every potential location of a recloser in the candidate primary sections, resulting in a decision matrix that is normalized and weighted using the CRITIC method to determine the optimal location based on a minimum criterion.

The analysis is repeated N times using the Monte Carlo method to generate scenarios, thereby establishing the probability of occurrence for each winning alternative. The final optimal location of the first recloser within the distribution system is selected. The methodology also includes switching coordination, repeating the analysis described earlier to determine the location of a second recloser, considering that the first recloser has already been identified as a winning alternative in generating numerous scenarios.

The proposed methodology applies to any distribution system and allows for analyzing multiple reclosers, considering cost constraints. Comprehensive programming is implemented in the Matlab software environment to carry out the analysis. The results successfully address maneuver and switching tests by minimizing all reliability indicators at the distribution level.

This methodology effectively resolves knowledge gaps in reliability studies by filling in data gaps using pseudo-random variable generation and identifying recloser locations that optimize all reliability criteria within the distribution system.

The reliability analysis is conducted in different levels, with level 1 focusing on generation sufficiency, level 2 considering generation and transmission line availability, and level 3, the main focus of this research, assessing the reliability in electric energy distribution. In level 3, the evaluation is performed on the distribution systems rather than individual users, as consumers are connected to various load points (transformers) that are affected differently by failures in the distribution system.

Due to this impact variation, it is not feasible to devise a single function for calculating reliability indicators across all distribution systems. The analysis heavily relies on the topology of the electric distribution network and how failures in different network elements affect each load point.

Advancements in information technologies have facilitated reliability evaluation using iterative and more complex mathematical techniques, improving prediction accuracy. One such technique is the well-known Monte Carlo method, which calculates the probability of a specific result by analyzing numerous randomly generated scenarios. Analyzing many scenarios makes the process more efficient and enables accurate determination of the probability of a particular scenario occurring.

Many researchers who have addressed reliability evaluation highlight the challenge posed by the availability of historical data required to establish failure rates and duration times for the elements constituting the distribution system. Due to the need for such data, many reliability studies resort to measures provided by manufacturers and typical behavioral ranges.

The proposed approach utilizes the Monte Carlo method, which generates pseudo-random numbers within predetermined ranges. These pseudo-random variables create multiple scenarios in which failure rates and durations for each system element are simulated. The Monte Carlo method provides a relative error of 1/N, where N represents the random variable's number of scenarios or components. It's important to note that the relative error decreases as the number of scenarios increases.

The Monte Carlo method is a powerful tool for this analysis as it allows for a more straightforward determination of the limits within which the analysis variables, such

as failure rates and durations of system elements, are likely to fall. This simplifies the central challenge of reliability analysis: the absence of precise data on these variables. The analysis can proceed effectively despite the lack of detailed data by utilizing the Monte Carlo method.

The multicriteria optimization technique is utilized to analyze the reliability indicators obtained for each considered scenario. The algorithm assigns specific locations to the switching devices (reclosers) in the candidate primary sections, excluding the final sections of the radial circuit that are already determined by switching load points (transformers).

The reliability indicators (SAIFI, SAIDI, CAIDI, AENS) are recalculated for each switching assignment, generating an individual scenario result for each objective criterion. This iterative process continues until all proposed candidate primary sections have been evaluated. As a result of this analysis, a discrete vector is obtained, containing solutions corresponding to each scenario.

The decision matrix is then constructed, with each reliability indicator represented in a column (criteria) for each solution vector of each criterion. The Exhaustive Search (Brute Force) algorithm considers all candidate scenarios (primary sections that do not terminate the circuit).

It guarantees the exploration of all feasible scenarios and ensures an optimal solution. Switching connection scenarios in the final sections of the distribution system are disregarded as they are already covered by the transformer switching. Tests have shown that the probability of these scenarios becoming the winning alternative is very low unless the failure rate in that section is exceptionally high and the load point has the highest number of users in the entire distribution system.

The number of scenarios determines the length of the decision matrix rows, where rows correspond to scenarios and columns represent criterion results. However, the decision matrix can be constructed row-wise or column-wise without affecting the subsequent analysis of weighted sums. Furthermore, computational time requirements are reasonable since this is a future planning problem for switching locations.

Establishing the decision matrix is essential to obtain individual results per scenario based on the optimal switching locations. The n rows of the decision matrix represent suitable alternatives among all switching location options, adhering to the criteria of being distinct and exclusive. The m columns represent the criteria (reliability indicators) as objective functions. The optimal option is selected based on the established decision criteria, eliminating any solutions inferior to others according to the dominance criterion.

The proposed methodology was evaluated as a case study using the 15-bus IEEE electric distribution test system operating at medium voltage. This test system was

obtained from the IEEE database. The system comprises 14 primary sections and 14 load points. Among these sections, section P1 already has a general recloser (R) that activates in the event of any interruption in the primary sections of that circuit.

Additionally, out of the 14 primary sections, the seven end sections (P4, P6, P7, P9, P12, P13, and P14) are excluded as candidate sections for recloser installation. Considering that a recloser is already installed in section P1 and that the end sections solely supply a load point with its protection at the transformer, the remaining candidate sections for recloser installation are P2, P3, P5, P8, P10, and P11. Extensive simulations have shown that the probability of these alternatives becoming the winning option could be much higher.

In this case study, the 14 load points (distribution transformers) are denoted by the letter (S) in the single-line diagram. Each load point is equipped with an independent switch that activates in case of the respective secondary system failures. The 15 buses of the system are numbered in the single-line diagram. Figure 3.9 illustrates the detailed single-line diagram based on the predefined conditions in the 15-bus IEEE test system.

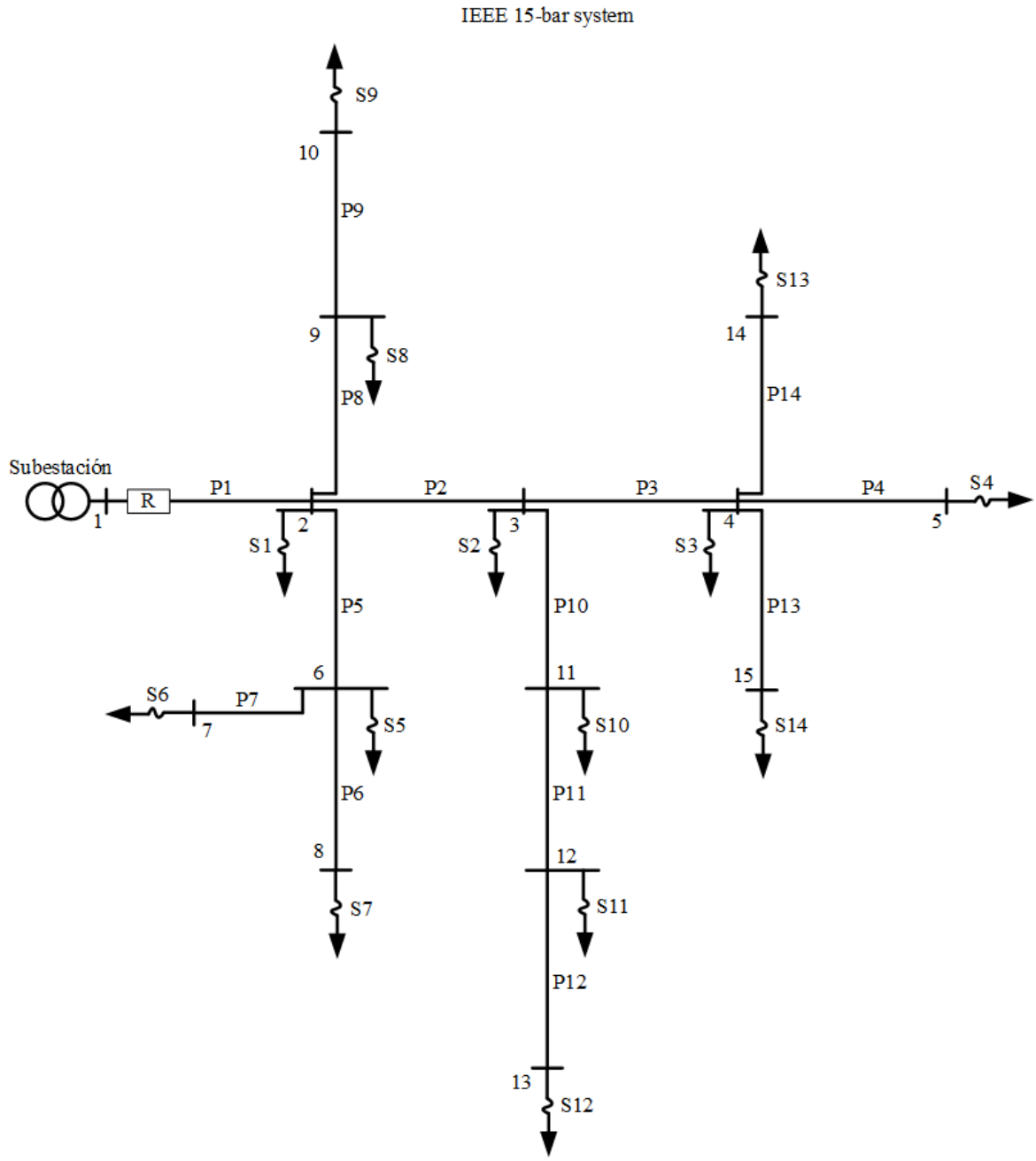


Figure 3.9: Graphical 15-bus IEEE test system.

The initial step involves establishing the random values for the base case and calculating the reliability indicators without considering any additional switching location apart from the existing one in P1. The obtained results for the pseudo-random analysis variables and reliability indicators will be presented for this base case.

In Figure 3.10, data of failure rates and duration times of these failures are displayed for each load point in the test distribution system. These data were generated using pseudo-random variables within defined and adjustable ranges.

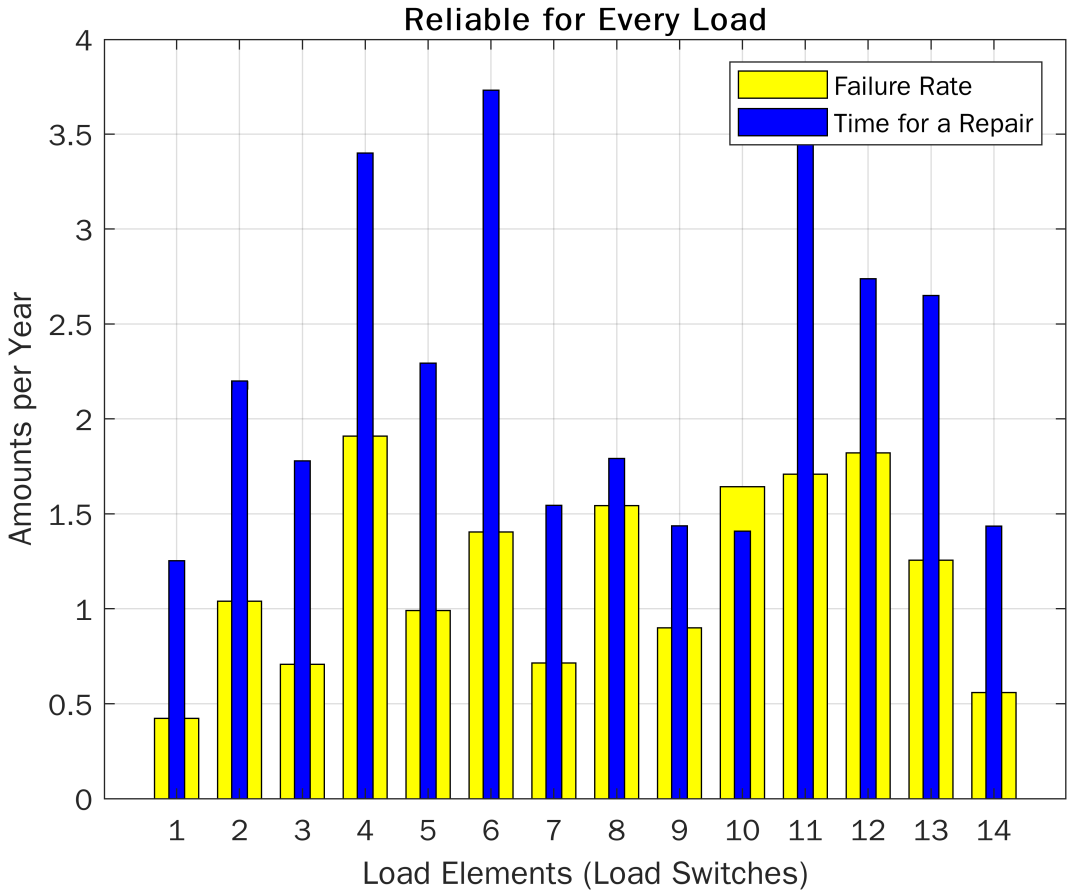


Figure 3.10: Fault data per year and fault repair times for each test system's load point (transformer).

Additionally, Figure 3.11 illustrates the failure rates of each primary element (sections P1, P2, P3, P4, P5, P6, P7, P8, P9, P10, P11, P12, P13, and P14), while Figure 3.12 presents the interruption repair times for these elements. These data were also generated using pseudo-random variables within adjustable ranges.

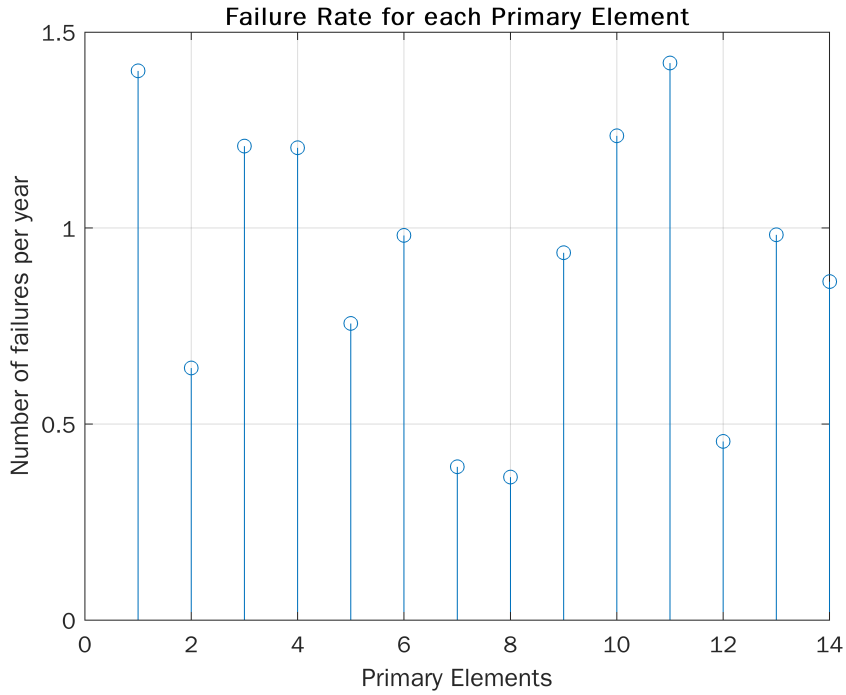


Figure 3.11: The failure rate for each primary element.

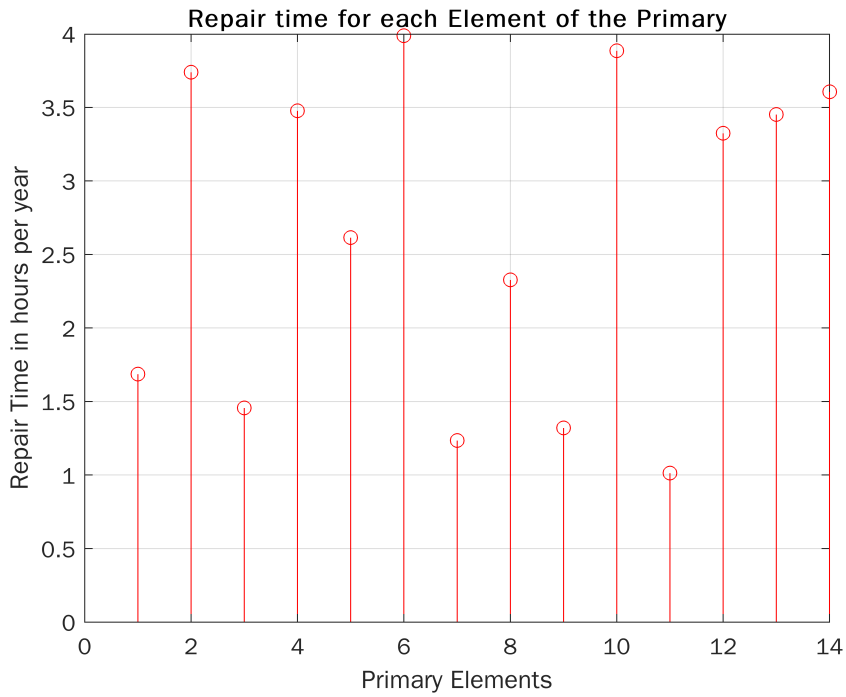


Figure 3.12: Repair time for each element of the primary.

The proposed example focuses on residential users; however, the model can

be adapted to incorporate accurate data if available. For the initial conditions of the problem, information regarding the number of users and their average consumption is known. Figure 3.13 provides details on the number of residential consumers connected to each transformer and the average power consumption at each load point (S1, S2, S3, S4, S5, S6, S7, S8, S9, S10, S11, S12, S13, and S14).

The analysis employs a counting technique to determine the potential occurrences of failures affecting each user (load point) based on the input data. It calculates the total number of failures per year that can impact a user connected to a specific transformer (load point), considering the failure rates at each load point and primary section. This information is visualized in Figure 3.14.

Failures affecting users can occur either through the operation of their transformer switch or the existing recloser (P1) interrupting the power supply to the entire circuit when a failure occurs in any primary section of the test system. Additionally, considering all potential failures in the design and their respective repair times, the total duration of losses experienced by an individual user throughout the year is determined by summing up the repair times of all system failures that could interrupt their service. Figure 3.15 provides an overview of the total duration of losses that may impact each user in a given year.

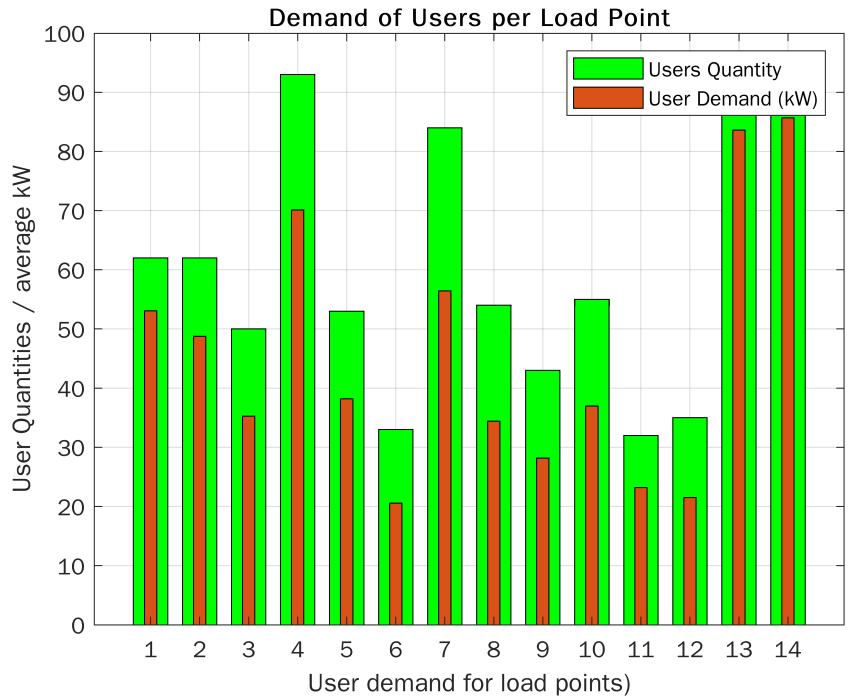


Figure 3.13: Number of power consumers per electric load point and average total consumption per load point.

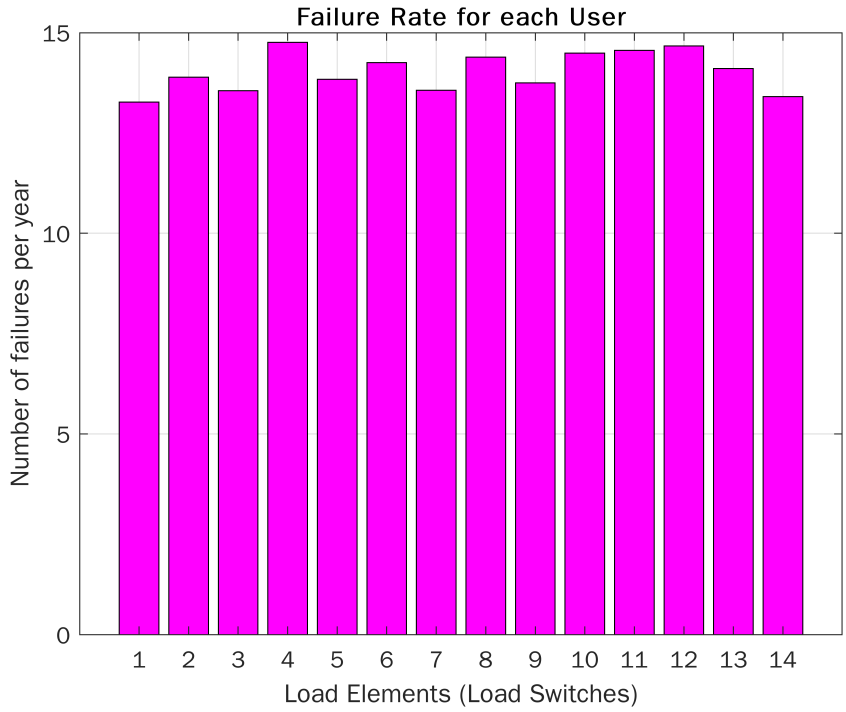


Figure 3.14: The total failure rate for each system user (load point).

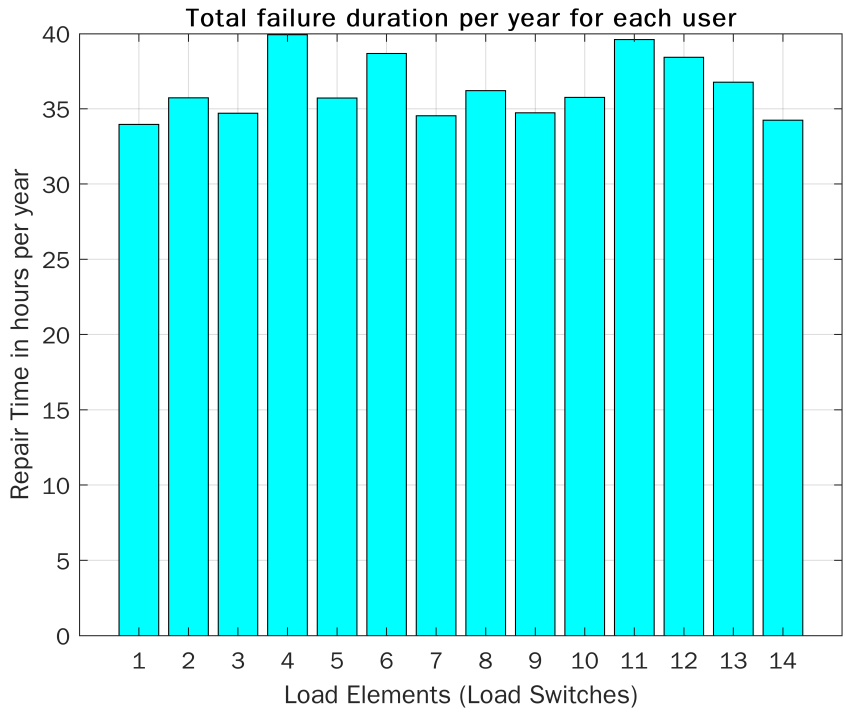


Figure 3.15: Total failure duration per year for each user.

By analyzing the data on total failure rates and the corresponding total duration of failures that may impact a specific user in the test distribution system over a year, we can calculate the average time of each user's system failure (load point). This calculation is derived from dividing the pseudo-random variables representing the total duration of failures by the total failure rate in a year per user. The average duration time for a single system failure for each user can be seen in Figure 3.16.

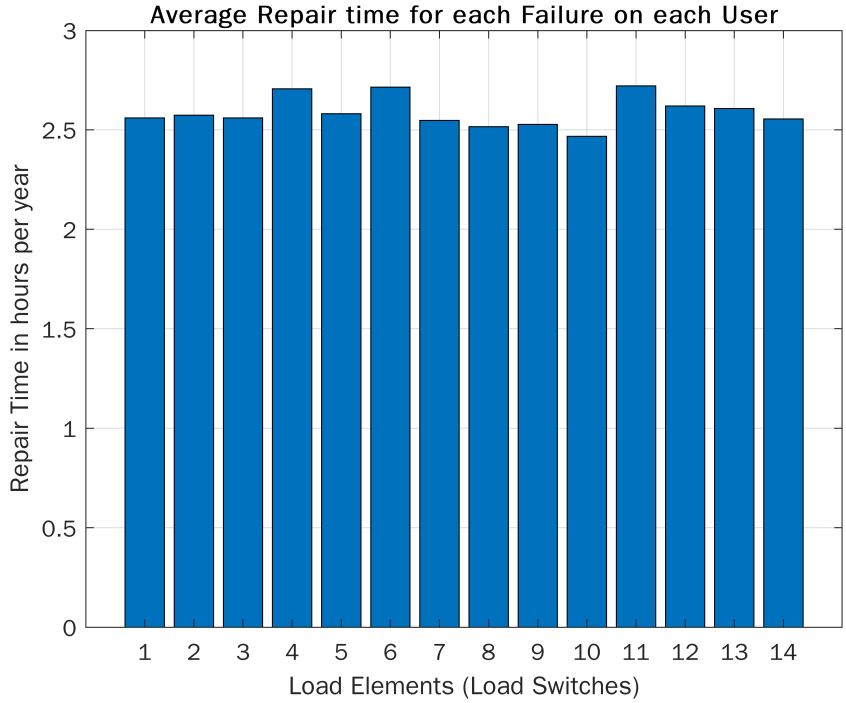


Figure 3.16: The average duration of a single failure for each system user.

By utilizing these given initial conditions, one can calculate the reliability indicators (SAIFI, SAIDI, CAIDI, and AENS) for the tested electric distribution system, assuming the presence of a lone recloser at P1 (4).

$$SAIFI = \sum_{i=1}^{n} \frac{\lambda i Ni}{Nt} \tag{3.13}$$

SAIFI represents the average number of outages experienced by each customer within a year in a specific study system, where λi denotes the failure rate, Ni represents the number of users per location, and Nt indicates the total number of users served. For the specific study case, the SAIFI value is 13,0018 yearly failures.

SAIDI: System Average Interruption Duration Index.

$$SAIDI = \sum_{i=1}^{n} \frac{UiNi}{Nt} \tag{3.14}$$

SAIDI is a reliability indicator that quantifies the average interruption time experienced by customers in a given location for a year. It is typically measured in time units, commonly hours. In the calculation of SAIDI, Ni represents the number of clients in location i, Ui represents the yearly interruption time for location i, and Nt indicates the total number of users served. For the specific study case, the SAIDI value is 31,0908 hours of failures per year.

CAIDI: Customer Average Interruption Duration Index.

$$CAIDI = \frac{\sum_{i=1}^{n} UiNi}{\sum_{i=1}^{n} \lambda iNi} = \frac{SAIDI}{SAIFI} \tag{3.15}$$

CAIDI is a reliability indicator that measures the average interruption duration experienced by any client in a given system. It is typically expressed in time units such as minutes or hours. CAIDI can be interpreted as the average restoration time, representing the time it takes to restore power to a customer after an interruption. For the specific study case, the CAIDI value is 2,3913 hours per failure.

AENS: Average Energy Not Supplied.

$$AENS = \sum_{i=1}^{n} kWiUi \tag{3.16}$$

AENS is a reliability index utilized in electric power systems to quantify the average energy not supplied per user. It measures the amount of energy, typically expressed in kilowatt-hours (kWh), that is not delivered to each customer on average. For the specific study case, the AENS value is 24,9538 kWh/year for each user (independent consumer) of the test system.

Based on the analysis of the four indicators, the findings indicate that the system lacks reliability. To address this issue, appropriate measures should be implemented to identify the optimal locations for additional switching. This strategic placement of switches would effectively reduce overall failure rates and system restoration times, thereby minimizing the values of the reliability indicators.

This optimization study will examine the optimal placement of an additional recloser and the one already present at (P1). The previous calculations will be simulated under various scenarios for multicriteria analysis. Each scenario represents the placement of an additional recloser in candidate primary sections that are not circuit end sections. The following scenarios will be considered: Scenario 1: Recloser at P2, Scenario 2: Recloser at P3, Scenario 3: Recloser at P5, Scenario 4: Recloser at P8, Scenario 5: Recloser at P10, and Scenario 6: Recloser at P11. These simulations aim to assess the impact of each scenario on the calculated values under the initial conditions.

The following algorithm enables the evaluation of reliability indicators in each proposed scenario and determines the optimal location using the multicriteria decision method described earlier.

```
clear all; clc;
tic
Barras=15; % Sistema IEEE de prueba
Barras_Carga=14;
Troncales_Primarios=14;
N=100000;
error=1/sqrt(N)
for x=1:1:N
Elementos=['S02';'S03';'S04';'S05';'S06';'S07';'S08';'S09';'S10';'
    S11';'S12';'S13';'S14';'S15';'P01';'P02';'P03';'P04';'P05';'P06';
    'P07';'P08';'P09';'P10';'P11';'P12';'P13';'P14'];
a = 0.3; % Minimum number of failures for Primary Trunk Elements
b = 1.5; % Maximum number of failures for Primary Trunk Elements
a1 = 0.4; % Minimum number of failures for charging points (
    transformers)
b1 = 2.3; % Maximum number of failures for charging points (
    transformers)
Tasa_Fallas_Troncales = (b-a).*rand(Troncales_Primarios,1) + a;
Tasa_Fallas_Cargas = (b1-a1).*rand(Barras_Carga,1) + a1;

c = 1; % Minimum repair time for Elements
d = 4; % Maximum repair time for Elements
r_Troncales = (d-c).*rand(length(Tasa_Fallas_Troncales),1) + c;
r_Cargas = (d-c).*rand(length(Tasa_Fallas_Cargas),1) + c;
U_Troncales=Tasa_Fallas_Troncales.*r_Troncales; % Duration of
    Interruptions by Primary Elements
U_Cargas=Tasa_Fallas_Cargas.*r_Cargas; % Duration of Interruptions
    by Load Elements
r_promedio=mean(U_Cargas./Tasa_Fallas_Cargas);

e = 0.6; % kW/Usuario (Minimum consumption)
f = 0.9; % kW/Usuario (Maximum consumption)
Rango_Consumo = (f-e).*rand(Barras_Carga,1) + e;
g = 25; % kW (Minimum number of Users per charging point)
h = 100; % kW (Maximum number of Users per charging point)
Cantidad_Usuarios=randi([g h],1,Barras_Carga)';
Demanda_Usuario = Cantidad_Usuarios.*Rango_Consumo;

Total_Consumidores=sum(Cantidad_Usuarios);

% Analysis of all scenarios for the connection of two additional
    reclosers (Exhaustive Search)
% Analysis with reclosers in P1, in P2 and in P3
Tasa_Fallas_P1_P2_P3_Usuario1=Tasa_Fallas_Cargas(1)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
Tasa_Fallas_P1_P2_P3_Usuario2=Tasa_Fallas_Cargas(2)+sum(
    Tasa_Fallas_Troncales(1:2))+sum(Tasa_Fallas_Troncales(5:12));
Tasa_Fallas_P1_P2_P3_Usuario3=Tasa_Fallas_Cargas(3)+sum(
```

```
    Tasa_Fallas_Troncales);
40  Tasa_Fallas_P1_P2_P3_Usuario4=Tasa_Fallas_Cargas(4)+sum(
    Tasa_Fallas_Troncales);
41  Tasa_Fallas_P1_P2_P3_Usuario5=Tasa_Fallas_Cargas(5)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
42  Tasa_Fallas_P1_P2_P3_Usuario6=Tasa_Fallas_Cargas(6)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
43  Tasa_Fallas_P1_P2_P3_Usuario7=Tasa_Fallas_Cargas(7)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
44  Tasa_Fallas_P1_P2_P3_Usuario8=Tasa_Fallas_Cargas(8)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
45  Tasa_Fallas_P1_P2_P3_Usuario9=Tasa_Fallas_Cargas(9)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
46  Tasa_Fallas_P1_P2_P3_Usuario10=Tasa_Fallas_Cargas(10)+sum(
    Tasa_Fallas_Troncales(1:2))+sum(Tasa_Fallas_Troncales(5:12));
47  Tasa_Fallas_P1_P2_P3_Usuario11=Tasa_Fallas_Cargas(11)+sum(
    Tasa_Fallas_Troncales(1:2))+sum(Tasa_Fallas_Troncales(5:12));
48  Tasa_Fallas_P1_P2_P3_Usuario12=Tasa_Fallas_Cargas(12)+sum(
    Tasa_Fallas_Troncales(1:2))+sum(Tasa_Fallas_Troncales(5:12));
49  Tasa_Fallas_P1_P2_P3_Usuario13=Tasa_Fallas_Cargas(13)+sum(
    Tasa_Fallas_Troncales);
50  Tasa_Fallas_P1_P2_P3_Usuario14=Tasa_Fallas_Cargas(14)+sum(
    Tasa_Fallas_Troncales);
51
52  Tasa_Fallas_P1_P2_P3_Usuarios=[Tasa_Fallas_P1_P2_P3_Usuario1
    Tasa_Fallas_P1_P2_P3_Usuario2 Tasa_Fallas_P1_P2_P3_Usuario3
    Tasa_Fallas_P1_P2_P3_Usuario4 Tasa_Fallas_P1_P2_P3_Usuario5
    Tasa_Fallas_P1_P2_P3_Usuario6 Tasa_Fallas_P1_P2_P3_Usuario7
    Tasa_Fallas_P1_P2_P3_Usuario8 Tasa_Fallas_P1_P2_P3_Usuario9
    Tasa_Fallas_P1_P2_P3_Usuario10 Tasa_Fallas_P1_P2_P3_Usuario11
    Tasa_Fallas_P1_P2_P3_Usuario12 Tasa_Fallas_P1_P2_P3_Usuario13
    Tasa_Fallas_P1_P2_P3_Usuario14];
53
54  Duracion_Reparacion_P1_P2_P3_Usuario1=U_Cargas(1)+U_Troncales(1)+sum
    (U_Troncales(5:9));
55  Duracion_Reparacion_P1_P2_P3_Usuario2=U_Cargas(2)+sum(U_Troncales
    (1:2))+sum(U_Troncales(5:12));
56  Duracion_Reparacion_P1_P2_P3_Usuario3=U_Cargas(3)+sum(U_Troncales);
57  Duracion_Reparacion_P1_P2_P3_Usuario4=U_Cargas(4)+sum(U_Troncales);
58  Duracion_Reparacion_P1_P2_P3_Usuario5=U_Cargas(5)+U_Troncales(1)+sum
    (U_Troncales(5:9));
59  Duracion_Reparacion_P1_P2_P3_Usuario6=U_Cargas(6)+U_Troncales(1)+sum
    (U_Troncales(5:9));
60  Duracion_Reparacion_P1_P2_P3_Usuario7=U_Cargas(7)+U_Troncales(1)+sum
    (U_Troncales(5:9));
61  Duracion_Reparacion_P1_P2_P3_Usuario8=U_Cargas(8)+U_Troncales(1)+sum
    (U_Troncales(5:9));
62  Duracion_Reparacion_P1_P2_P3_Usuario9=U_Cargas(9)+U_Troncales(1)+sum
    (U_Troncales(5:9));
63  Duracion_Reparacion_P1_P2_P3_Usuario10=U_Cargas(10)+sum(U_Troncales
    (1:2))+sum(U_Troncales(5:12));
64  Duracion_Reparacion_P1_P2_P3_Usuario11=U_Cargas(11)+sum(U_Troncales
```

```
    (1:2))+sum(U_Troncales(5:12));
65  Duracion_Reparacion_P1_P2_P3_Usuario12=U_Cargas(12)+sum(U_Troncales
    (1:2))+sum(U_Troncales(5:12));
66  Duracion_Reparacion_P1_P2_P3_Usuario13=U_Cargas(13)+sum(U_Troncales)
    ;
67  Duracion_Reparacion_P1_P2_P3_Usuario14=U_Cargas(14)+sum(U_Troncales)
    ;
68
69  Duracion_Reparacion_P1_P2_P3_Usuarios=[
    Duracion_Reparacion_P1_P2_P3_Usuario1
    Duracion_Reparacion_P1_P2_P3_Usuario2
    Duracion_Reparacion_P1_P2_P3_Usuario3
    Duracion_Reparacion_P1_P2_P3_Usuario4
    Duracion_Reparacion_P1_P2_P3_Usuario5
    Duracion_Reparacion_P1_P2_P3_Usuario6
    Duracion_Reparacion_P1_P2_P3_Usuario7
    Duracion_Reparacion_P1_P2_P3_Usuario8
    Duracion_Reparacion_P1_P2_P3_Usuario9
    Duracion_Reparacion_P1_P2_P3_Usuario10
    Duracion_Reparacion_P1_P2_P3_Usuario11
    Duracion_Reparacion_P1_P2_P3_Usuario12
    Duracion_Reparacion_P1_P2_P3_Usuario13
    Duracion_Reparacion_P1_P2_P3_Usuario14];
70
71  SAIFI_P1_P2_P3 = (Tasa_Fallas_P1_P2_P3_Usuario1*Cantidad_Usuarios(1)
    +Tasa_Fallas_P1_P2_P3_Usuario2*Cantidad_Usuarios(2)+
    Tasa_Fallas_P1_P2_P3_Usuario3*Cantidad_Usuarios(3)+
    Tasa_Fallas_P1_P2_P3_Usuario4*Cantidad_Usuarios(4)+
    Tasa_Fallas_P1_P2_P3_Usuario5*Cantidad_Usuarios(5)+
    Tasa_Fallas_P1_P2_P3_Usuario6*Cantidad_Usuarios(6)+
    Tasa_Fallas_P1_P2_P3_Usuario7*Cantidad_Usuarios(7)+
    Tasa_Fallas_P1_P2_P3_Usuario8*Cantidad_Usuarios(8)+
    Tasa_Fallas_P1_P2_P3_Usuario9*Cantidad_Usuarios(9)+
    Tasa_Fallas_P1_P2_P3_Usuario10*Cantidad_Usuarios(10)+
    Tasa_Fallas_P1_P2_P3_Usuario11*Cantidad_Usuarios(11)+
    Tasa_Fallas_P1_P2_P3_Usuario12*Cantidad_Usuarios(12)+
    Tasa_Fallas_P1_P2_P3_Usuario13*Cantidad_Usuarios(13)+
    Tasa_Fallas_P1_P2_P3_Usuario14*Cantidad_Usuarios(14))/
    Total_Consumidores; % Number of average failures per year per
    user
72  SAIDI_P1_P2_P3 = (Duracion_Reparacion_P1_P2_P3_Usuario1*
    Cantidad_Usuarios(1)+Duracion_Reparacion_P1_P2_P3_Usuario2*
    Cantidad_Usuarios(2)+Duracion_Reparacion_P1_P2_P3_Usuario3*
    Cantidad_Usuarios(3)+Duracion_Reparacion_P1_P2_P3_Usuario4*
    Cantidad_Usuarios(4)+Duracion_Reparacion_P1_P2_P3_Usuario5*
    Cantidad_Usuarios(5)+Duracion_Reparacion_P1_P2_P3_Usuario6*
    Cantidad_Usuarios(6)+Duracion_Reparacion_P1_P2_P3_Usuario7*
    Cantidad_Usuarios(7)+Duracion_Reparacion_P1_P2_P3_Usuario8*
    Cantidad_Usuarios(8)+Duracion_Reparacion_P1_P2_P3_Usuario9*
    Cantidad_Usuarios(9)+Duracion_Reparacion_P1_P2_P3_Usuario10*
    Cantidad_Usuarios(10)+Duracion_Reparacion_P1_P2_P3_Usuario11*
    Cantidad_Usuarios(11)+Duracion_Reparacion_P1_P2_P3_Usuario12*
```

```
    Cantidad_Usuarios(12)+Duracion_Reparacion_P1_P2_P3_Usuario13*
    Cantidad_Usuarios(13)+Duracion_Reparacion_P1_P2_P3_Usuario14*
    Cantidad_Usuarios(14))/Total_Consumidores; % Average repair time
    per year per user
CAIDI_P1_P2_P3 = SAIDI_P1_P2_P3 / SAIFI_P1_P2_P3; % Average Duration
     of an Interruption
AENS_P1_P2_P3 = (Duracion_Reparacion_P1_P2_P3_Usuario1*
    Demanda_Usuario(1)+Duracion_Reparacion_P1_P2_P3_Usuario2*
    Demanda_Usuario(2)+Duracion_Reparacion_P1_P2_P3_Usuario3*
    Demanda_Usuario(3)+Duracion_Reparacion_P1_P2_P3_Usuario4*
    Demanda_Usuario(4)+Duracion_Reparacion_P1_P2_P3_Usuario5*
    Demanda_Usuario(5)+Duracion_Reparacion_P1_P2_P3_Usuario6*
    Demanda_Usuario(6)+Duracion_Reparacion_P1_P2_P3_Usuario7*
    Demanda_Usuario(7)+Duracion_Reparacion_P1_P2_P3_Usuario8*
    Demanda_Usuario(8)+Duracion_Reparacion_P1_P2_P3_Usuario9*
    Demanda_Usuario(9)+Duracion_Reparacion_P1_P2_P3_Usuario10*
    Demanda_Usuario(10)+Duracion_Reparacion_P1_P2_P3_Usuario11*
    Demanda_Usuario(11)+Duracion_Reparacion_P1_P2_P3_Usuario12*
    Demanda_Usuario(12)+Duracion_Reparacion_P1_P2_P3_Usuario13*
    Demanda_Usuario(13)+Duracion_Reparacion_P1_P2_P3_Usuario14*
    Demanda_Usuario(14))/Total_Consumidores; % Average energy NOT
    supplied per year per user (kWh Not Supplied / User)


% Analysis with reclosers in P1, in P2 and in P5
Tasa_Fallas_P1_P2_P5_Usuario1=Tasa_Fallas_Cargas(1)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(8:9));
Tasa_Fallas_P1_P2_P5_Usuario2=Tasa_Fallas_Cargas(2)+sum(
    Tasa_Fallas_Troncales(1:4))+sum(Tasa_Fallas_Troncales(8:14));
Tasa_Fallas_P1_P2_P5_Usuario3=Tasa_Fallas_Cargas(3)+sum(
    Tasa_Fallas_Troncales(1:4))+sum(Tasa_Fallas_Troncales(8:14));
Tasa_Fallas_P1_P2_P5_Usuario4=Tasa_Fallas_Cargas(4)+sum(
    Tasa_Fallas_Troncales(1:4))+sum(Tasa_Fallas_Troncales(8:14));
Tasa_Fallas_P1_P2_P5_Usuario5=Tasa_Fallas_Cargas(5)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
Tasa_Fallas_P1_P2_P5_Usuario6=Tasa_Fallas_Cargas(6)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
Tasa_Fallas_P1_P2_P5_Usuario7=Tasa_Fallas_Cargas(7)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
Tasa_Fallas_P1_P2_P5_Usuario8=Tasa_Fallas_Cargas(8)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(8:9));
Tasa_Fallas_P1_P2_P5_Usuario9=Tasa_Fallas_Cargas(9)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(8:9));
Tasa_Fallas_P1_P2_P5_Usuario10=Tasa_Fallas_Cargas(10)+sum(
    Tasa_Fallas_Troncales(1:4))+sum(Tasa_Fallas_Troncales(8:14));
Tasa_Fallas_P1_P2_P5_Usuario11=Tasa_Fallas_Cargas(11)+sum(
    Tasa_Fallas_Troncales(1:4))+sum(Tasa_Fallas_Troncales(8:14));
Tasa_Fallas_P1_P2_P5_Usuario12=Tasa_Fallas_Cargas(12)+sum(
    Tasa_Fallas_Troncales(1:4))+sum(Tasa_Fallas_Troncales(8:14));
Tasa_Fallas_P1_P2_P5_Usuario13=Tasa_Fallas_Cargas(13)+sum(
    Tasa_Fallas_Troncales(1:4))+sum(Tasa_Fallas_Troncales(8:14));
Tasa_Fallas_P1_P2_P5_Usuario14=Tasa_Fallas_Cargas(14)+sum(
```

```
    Tasa_Fallas_Troncales(1:4))+sum(Tasa_Fallas_Troncales(8:14));
92
93  Tasa_Fallas_P1_P2_P5_Usuarios=[Tasa_Fallas_P1_P2_P5_Usuario1
    Tasa_Fallas_P1_P2_P5_Usuario2 Tasa_Fallas_P1_P2_P5_Usuario3
    Tasa_Fallas_P1_P2_P5_Usuario4 Tasa_Fallas_P1_P2_P5_Usuario5
    Tasa_Fallas_P1_P2_P5_Usuario6 Tasa_Fallas_P1_P2_P5_Usuario7
    Tasa_Fallas_P1_P2_P5_Usuario8 Tasa_Fallas_P1_P2_P5_Usuario9
    Tasa_Fallas_P1_P2_P5_Usuario10 Tasa_Fallas_P1_P2_P5_Usuario11
    Tasa_Fallas_P1_P2_P5_Usuario12 Tasa_Fallas_P1_P2_P5_Usuario13
    Tasa_Fallas_P1_P2_P5_Usuario14];
94
95  Duracion_Reparacion_P1_P2_P5_Usuario1=U_Cargas(1)+U_Troncales(1)+sum
    (U_Troncales(8:9));
96  Duracion_Reparacion_P1_P2_P5_Usuario2=U_Cargas(2)+sum(U_Troncales
    (1:4))+sum(U_Troncales(8:14));
97  Duracion_Reparacion_P1_P2_P5_Usuario3=U_Cargas(3)+sum(U_Troncales
    (1:4))+sum(U_Troncales(8:14));
98  Duracion_Reparacion_P1_P2_P5_Usuario4=U_Cargas(4)+sum(U_Troncales
    (1:4))+sum(U_Troncales(8:14));
99  Duracion_Reparacion_P1_P2_P5_Usuario5=U_Cargas(5)+U_Troncales(1)+sum
    (U_Troncales(5:9));
100 Duracion_Reparacion_P1_P2_P5_Usuario6=U_Cargas(6)+U_Troncales(1)+sum
    (U_Troncales(5:9));
101 Duracion_Reparacion_P1_P2_P5_Usuario7=U_Cargas(7)+U_Troncales(1)+sum
    (U_Troncales(5:9));
102 Duracion_Reparacion_P1_P2_P5_Usuario8=U_Cargas(8)+U_Troncales(1)+sum
    (U_Troncales(8:9));
103 Duracion_Reparacion_P1_P2_P5_Usuario9=U_Cargas(9)+U_Troncales(1)+sum
    (U_Troncales(8:9));
104 Duracion_Reparacion_P1_P2_P5_Usuario10=U_Cargas(10)+sum(U_Troncales
    (1:4))+sum(U_Troncales(8:14));
105 Duracion_Reparacion_P1_P2_P5_Usuario11=U_Cargas(11)+sum(U_Troncales
    (1:4))+sum(U_Troncales(8:14));
106 Duracion_Reparacion_P1_P2_P5_Usuario12=U_Cargas(12)+sum(U_Troncales
    (1:4))+sum(U_Troncales(8:14));
107 Duracion_Reparacion_P1_P2_P5_Usuario13=U_Cargas(13)+sum(U_Troncales
    (1:4))+sum(U_Troncales(8:14));
108 Duracion_Reparacion_P1_P2_P5_Usuario14=U_Cargas(14)+sum(U_Troncales
    (1:4))+sum(U_Troncales(8:14));
109
110 Duracion_Reparacion_P1_P2_P5_Usuarios=[
    Duracion_Reparacion_P1_P2_P5_Usuario1
    Duracion_Reparacion_P1_P2_P5_Usuario2
    Duracion_Reparacion_P1_P2_P5_Usuario3
    Duracion_Reparacion_P1_P2_P5_Usuario4
    Duracion_Reparacion_P1_P2_P5_Usuario5
    Duracion_Reparacion_P1_P2_P5_Usuario6
    Duracion_Reparacion_P1_P2_P5_Usuario7
    Duracion_Reparacion_P1_P2_P5_Usuario8
    Duracion_Reparacion_P1_P2_P5_Usuario9
    Duracion_Reparacion_P1_P2_P5_Usuario10
    Duracion_Reparacion_P1_P2_P5_Usuario11
```

```
    Duracion_Reparacion_P1_P2_P5_Usuario12
    Duracion_Reparacion_P1_P2_P5_Usuario13
    Duracion_Reparacion_P1_P2_P5_Usuario14];
111
112 SAIFI_P1_P2_P5 = (Tasa_Fallas_P1_P2_P5_Usuario1*Cantidad_Usuarios(1)
    +Tasa_Fallas_P1_P2_P5_Usuario2*Cantidad_Usuarios(2)+
    Tasa_Fallas_P1_P2_P5_Usuario3*Cantidad_Usuarios(3)+
    Tasa_Fallas_P1_P2_P5_Usuario4*Cantidad_Usuarios(4)+
    Tasa_Fallas_P1_P2_P5_Usuario5*Cantidad_Usuarios(5)+
    Tasa_Fallas_P1_P2_P5_Usuario6*Cantidad_Usuarios(6)+
    Tasa_Fallas_P1_P2_P5_Usuario7*Cantidad_Usuarios(7)+
    Tasa_Fallas_P1_P2_P5_Usuario8*Cantidad_Usuarios(8)+
    Tasa_Fallas_P1_P2_P5_Usuario9*Cantidad_Usuarios(9)+
    Tasa_Fallas_P1_P2_P5_Usuario10*Cantidad_Usuarios(10)+
    Tasa_Fallas_P1_P2_P5_Usuario11*Cantidad_Usuarios(11)+
    Tasa_Fallas_P1_P2_P5_Usuario12*Cantidad_Usuarios(12)+
    Tasa_Fallas_P1_P2_P5_Usuario13*Cantidad_Usuarios(13)+
    Tasa_Fallas_P1_P2_P5_Usuario14*Cantidad_Usuarios(14))/
    Total_Consumidores; % Number of average failures per year per
    user
113 SAIDI_P1_P2_P5 = (Duracion_Reparacion_P1_P2_P5_Usuario1*
    Cantidad_Usuarios(1)+Duracion_Reparacion_P1_P2_P5_Usuario2*
    Cantidad_Usuarios(2)+Duracion_Reparacion_P1_P2_P5_Usuario3*
    Cantidad_Usuarios(3)+Duracion_Reparacion_P1_P2_P5_Usuario4*
    Cantidad_Usuarios(4)+Duracion_Reparacion_P1_P2_P5_Usuario5*
    Cantidad_Usuarios(5)+Duracion_Reparacion_P1_P2_P5_Usuario6*
    Cantidad_Usuarios(6)+Duracion_Reparacion_P1_P2_P5_Usuario7*
    Cantidad_Usuarios(7)+Duracion_Reparacion_P1_P2_P5_Usuario8*
    Cantidad_Usuarios(8)+Duracion_Reparacion_P1_P2_P5_Usuario9*
    Cantidad_Usuarios(9)+Duracion_Reparacion_P1_P2_P5_Usuario10*
    Cantidad_Usuarios(10)+Duracion_Reparacion_P1_P2_P5_Usuario11*
    Cantidad_Usuarios(11)+Duracion_Reparacion_P1_P2_P5_Usuario12*
    Cantidad_Usuarios(12)+Duracion_Reparacion_P1_P2_P5_Usuario13*
    Cantidad_Usuarios(13)+Duracion_Reparacion_P1_P2_P5_Usuario14*
    Cantidad_Usuarios(14))/Total_Consumidores; % Average repair time
    per year per user
114 CAIDI_P1_P2_P5 = SAIDI_P1_P2_P5 / SAIFI_P1_P2_P5; % Average Duration
     of an Interruption
115 AENS_P1_P2_P5 = (Duracion_Reparacion_P1_P2_P5_Usuario1*
    Demanda_Usuario(1)+Duracion_Reparacion_P1_P2_P5_Usuario2*
    Demanda_Usuario(2)+Duracion_Reparacion_P1_P2_P5_Usuario3*
    Demanda_Usuario(3)+Duracion_Reparacion_P1_P2_P5_Usuario4*
    Demanda_Usuario(4)+Duracion_Reparacion_P1_P2_P5_Usuario5*
    Demanda_Usuario(5)+Duracion_Reparacion_P1_P2_P5_Usuario6*
    Demanda_Usuario(6)+Duracion_Reparacion_P1_P2_P5_Usuario7*
    Demanda_Usuario(7)+Duracion_Reparacion_P1_P2_P5_Usuario8*
    Demanda_Usuario(8)+Duracion_Reparacion_P1_P2_P5_Usuario9*
    Demanda_Usuario(9)+Duracion_Reparacion_P1_P2_P5_Usuario10*
    Demanda_Usuario(10)+Duracion_Reparacion_P1_P2_P5_Usuario11*
    Demanda_Usuario(11)+Duracion_Reparacion_P1_P2_P5_Usuario12*
    Demanda_Usuario(12)+Duracion_Reparacion_P1_P2_P5_Usuario13*
    Demanda_Usuario(13)+Duracion_Reparacion_P1_P2_P5_Usuario14*
```

```
    Demanda_Usuario(14))/Total_Consumidores; % Average energy NOT
    supplied per year per user (kWh Not Supplied / User)
116
117 % Analysis with reclosers in P1, in P2 and in P8
118 Tasa_Fallas_P1_P2_P8_Usuario1=Tasa_Fallas_Cargas(1)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:7));
119 Tasa_Fallas_P1_P2_P8_Usuario2=Tasa_Fallas_Cargas(2)+sum(
    Tasa_Fallas_Troncales(1:7))+sum(Tasa_Fallas_Troncales(10:14));
120 Tasa_Fallas_P1_P2_P8_Usuario3=Tasa_Fallas_Cargas(3)+sum(
    Tasa_Fallas_Troncales(1:7))+sum(Tasa_Fallas_Troncales(10:14));
121 Tasa_Fallas_P1_P2_P8_Usuario4=Tasa_Fallas_Cargas(4)+sum(
    Tasa_Fallas_Troncales(1:7))+sum(Tasa_Fallas_Troncales(10:14));
122 Tasa_Fallas_P1_P2_P8_Usuario5=Tasa_Fallas_Cargas(5)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:7));
123 Tasa_Fallas_P1_P2_P8_Usuario6=Tasa_Fallas_Cargas(6)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:7));
124 Tasa_Fallas_P1_P2_P8_Usuario7=Tasa_Fallas_Cargas(7)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:7));
125 Tasa_Fallas_P1_P2_P8_Usuario8=Tasa_Fallas_Cargas(8)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(8:9));
126 Tasa_Fallas_P1_P2_P8_Usuario9=Tasa_Fallas_Cargas(9)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(8:9));
127 Tasa_Fallas_P1_P2_P8_Usuario10=Tasa_Fallas_Cargas(10)+sum(
    Tasa_Fallas_Troncales(1:7))+sum(Tasa_Fallas_Troncales(10:14));
128 Tasa_Fallas_P1_P2_P8_Usuario11=Tasa_Fallas_Cargas(11)+sum(
    Tasa_Fallas_Troncales(1:7))+sum(Tasa_Fallas_Troncales(10:14));
129 Tasa_Fallas_P1_P2_P8_Usuario12=Tasa_Fallas_Cargas(12)+sum(
    Tasa_Fallas_Troncales(1:7))+sum(Tasa_Fallas_Troncales(10:14));
130 Tasa_Fallas_P1_P2_P8_Usuario13=Tasa_Fallas_Cargas(13)+sum(
    Tasa_Fallas_Troncales(1:7))+sum(Tasa_Fallas_Troncales(10:14));
131 Tasa_Fallas_P1_P2_P8_Usuario14=Tasa_Fallas_Cargas(14)+sum(
    Tasa_Fallas_Troncales(1:7))+sum(Tasa_Fallas_Troncales(10:14));
132
133 Tasa_Fallas_P1_P2_P8_Usuarios=[Tasa_Fallas_P1_P2_P8_Usuario1
    Tasa_Fallas_P1_P2_P8_Usuario2 Tasa_Fallas_P1_P2_P8_Usuario3
    Tasa_Fallas_P1_P2_P8_Usuario4 Tasa_Fallas_P1_P2_P8_Usuario5
    Tasa_Fallas_P1_P2_P8_Usuario6 Tasa_Fallas_P1_P2_P8_Usuario7
    Tasa_Fallas_P1_P2_P8_Usuario8 Tasa_Fallas_P1_P2_P8_Usuario9
    Tasa_Fallas_P1_P2_P8_Usuario10 Tasa_Fallas_P1_P2_P8_Usuario11
    Tasa_Fallas_P1_P2_P8_Usuario12 Tasa_Fallas_P1_P2_P8_Usuario13
    Tasa_Fallas_P1_P2_P8_Usuario14];
134
135 Duracion_Reparacion_P1_P2_P8_Usuario1=U_Cargas(1)+U_Troncales(1)+sum
    (U_Troncales(5:7));
136 Duracion_Reparacion_P1_P2_P8_Usuario2=U_Cargas(2)+sum(U_Troncales
    (1:7))+sum(U_Troncales(10:14));
137 Duracion_Reparacion_P1_P2_P8_Usuario3=U_Cargas(3)+sum(U_Troncales
    (1:7))+sum(U_Troncales(10:14));
138 Duracion_Reparacion_P1_P2_P8_Usuario4=U_Cargas(4)+sum(U_Troncales
    (1:7))+sum(U_Troncales(10:14));
139 Duracion_Reparacion_P1_P2_P8_Usuario5=U_Cargas(5)+U_Troncales(1)+sum
    (U_Troncales(5:7));
```

```
140 Duracion_Reparacion_P1_P2_P8_Usuario6=U_Cargas(6)+U_Troncales(1)+sum
    (U_Troncales(5:7));
141 Duracion_Reparacion_P1_P2_P8_Usuario7=U_Cargas(7)+U_Troncales(1)+sum
    (U_Troncales(5:7));
142 Duracion_Reparacion_P1_P2_P8_Usuario8=U_Cargas(8)+U_Troncales(1)+sum
    (U_Troncales(8:9));
143 Duracion_Reparacion_P1_P2_P8_Usuario9=U_Cargas(9)+U_Troncales(1)+sum
    (U_Troncales(8:9));
144 Duracion_Reparacion_P1_P2_P8_Usuario10=U_Cargas(10)+sum(U_Troncales
    (1:7))+sum(U_Troncales(10:14));
145 Duracion_Reparacion_P1_P2_P8_Usuario11=U_Cargas(11)+sum(U_Troncales
    (1:7))+sum(U_Troncales(10:14));
146 Duracion_Reparacion_P1_P2_P8_Usuario12=U_Cargas(12)+sum(U_Troncales
    (1:7))+sum(U_Troncales(10:14));
147 Duracion_Reparacion_P1_P2_P8_Usuario13=U_Cargas(13)+sum(U_Troncales
    (1:7))+sum(U_Troncales(10:14));
148 Duracion_Reparacion_P1_P2_P8_Usuario14=U_Cargas(14)+sum(U_Troncales
    (1:7))+sum(U_Troncales(10:14));
149
150 Duracion_Reparacion_P1_P2_P8_Usuarios=[
    Duracion_Reparacion_P1_P2_P8_Usuario1
    Duracion_Reparacion_P1_P2_P8_Usuario2
    Duracion_Reparacion_P1_P2_P8_Usuario3
    Duracion_Reparacion_P1_P2_P8_Usuario4
    Duracion_Reparacion_P1_P2_P8_Usuario5
    Duracion_Reparacion_P1_P2_P8_Usuario6
    Duracion_Reparacion_P1_P2_P8_Usuario7
    Duracion_Reparacion_P1_P2_P8_Usuario8
    Duracion_Reparacion_P1_P2_P8_Usuario9
    Duracion_Reparacion_P1_P2_P8_Usuario10
    Duracion_Reparacion_P1_P2_P8_Usuario11
    Duracion_Reparacion_P1_P2_P8_Usuario12
    Duracion_Reparacion_P1_P2_P8_Usuario13
    Duracion_Reparacion_P1_P2_P8_Usuario14];
151
152 SAIFI_P1_P2_P8 = (Tasa_Fallas_P1_P2_P8_Usuario1*Cantidad_Usuarios(1)
    +Tasa_Fallas_P1_P2_P8_Usuario2*Cantidad_Usuarios(2)+
    Tasa_Fallas_P1_P2_P8_Usuario3*Cantidad_Usuarios(3)+
    Tasa_Fallas_P1_P2_P8_Usuario4*Cantidad_Usuarios(4)+
    Tasa_Fallas_P1_P2_P8_Usuario5*Cantidad_Usuarios(5)+
    Tasa_Fallas_P1_P2_P8_Usuario6*Cantidad_Usuarios(6)+
    Tasa_Fallas_P1_P2_P8_Usuario7*Cantidad_Usuarios(7)+
    Tasa_Fallas_P1_P2_P8_Usuario8*Cantidad_Usuarios(8)+
    Tasa_Fallas_P1_P2_P8_Usuario9*Cantidad_Usuarios(9)+
    Tasa_Fallas_P1_P2_P8_Usuario10*Cantidad_Usuarios(10)+
    Tasa_Fallas_P1_P2_P8_Usuario11*Cantidad_Usuarios(11)+
    Tasa_Fallas_P1_P2_P8_Usuario12*Cantidad_Usuarios(12)+
    Tasa_Fallas_P1_P2_P8_Usuario13*Cantidad_Usuarios(13)+
    Tasa_Fallas_P1_P2_P8_Usuario14*Cantidad_Usuarios(14))/
    Total_Consumidores; % Number of average failures per year per
    user
153 SAIDI_P1_P2_P8 = (Duracion_Reparacion_P1_P2_P8_Usuario1*
```

```
    Cantidad_Usuarios(1)+Duracion_Reparacion_P1_P2_P8_Usuario2*
    Cantidad_Usuarios(2)+Duracion_Reparacion_P1_P2_P8_Usuario3*
    Cantidad_Usuarios(3)+Duracion_Reparacion_P1_P2_P8_Usuario4*
    Cantidad_Usuarios(4)+Duracion_Reparacion_P1_P2_P8_Usuario5*
    Cantidad_Usuarios(5)+Duracion_Reparacion_P1_P2_P8_Usuario6*
    Cantidad_Usuarios(6)+Duracion_Reparacion_P1_P2_P8_Usuario7*
    Cantidad_Usuarios(7)+Duracion_Reparacion_P1_P2_P8_Usuario8*
    Cantidad_Usuarios(8)+Duracion_Reparacion_P1_P2_P8_Usuario9*
    Cantidad_Usuarios(9)+Duracion_Reparacion_P1_P2_P8_Usuario10*
    Cantidad_Usuarios(10)+Duracion_Reparacion_P1_P2_P8_Usuario11*
    Cantidad_Usuarios(11)+Duracion_Reparacion_P1_P2_P8_Usuario12*
    Cantidad_Usuarios(12)+Duracion_Reparacion_P1_P2_P8_Usuario13*
    Cantidad_Usuarios(13)+Duracion_Reparacion_P1_P2_P8_Usuario14*
    Cantidad_Usuarios(14))/Total_Consumidores; % Average repair time
    per year per user
154 CAIDI_P1_P2_P8 = SAIDI_P1_P2_P8 / SAIFI_P1_P2_P8; % Average Duration
     of an Interruption
155 AENS_P1_P2_P8 = (Duracion_Reparacion_P1_P2_P8_Usuario1*
    Demanda_Usuario(1)+Duracion_Reparacion_P1_P2_P8_Usuario2*
    Demanda_Usuario(2)+Duracion_Reparacion_P1_P2_P8_Usuario3*
    Demanda_Usuario(3)+Duracion_Reparacion_P1_P2_P8_Usuario4*
    Demanda_Usuario(4)+Duracion_Reparacion_P1_P2_P8_Usuario5*
    Demanda_Usuario(5)+Duracion_Reparacion_P1_P2_P8_Usuario6*
    Demanda_Usuario(6)+Duracion_Reparacion_P1_P2_P8_Usuario7*
    Demanda_Usuario(7)+Duracion_Reparacion_P1_P2_P8_Usuario8*
    Demanda_Usuario(8)+Duracion_Reparacion_P1_P2_P8_Usuario9*
    Demanda_Usuario(9)+Duracion_Reparacion_P1_P2_P8_Usuario10*
    Demanda_Usuario(10)+Duracion_Reparacion_P1_P2_P8_Usuario11*
    Demanda_Usuario(11)+Duracion_Reparacion_P1_P2_P8_Usuario12*
    Demanda_Usuario(12)+Duracion_Reparacion_P1_P2_P8_Usuario13*
    Demanda_Usuario(13)+Duracion_Reparacion_P1_P2_P8_Usuario14*
    Demanda_Usuario(14))/Total_Consumidores; % Average energy NOT
    supplied per year per user (kWh Not Supplied / User)
156
157
158 % Analysis with reclosers in P1, in P2 and in P10
159 Tasa_Fallas_P1_P2_P10_Usuario1=Tasa_Fallas_Cargas(1)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
160 Tasa_Fallas_P1_P2_P10_Usuario2=Tasa_Fallas_Cargas(2)+sum(
    Tasa_Fallas_Troncales(1:9))+sum(Tasa_Fallas_Troncales(13:14));
161 Tasa_Fallas_P1_P2_P10_Usuario3=Tasa_Fallas_Cargas(3)+sum(
    Tasa_Fallas_Troncales(1:9))+sum(Tasa_Fallas_Troncales(13:14));
162 Tasa_Fallas_P1_P2_P10_Usuario4=Tasa_Fallas_Cargas(4)+sum(
    Tasa_Fallas_Troncales(1:9))+sum(Tasa_Fallas_Troncales(13:14));
163 Tasa_Fallas_P1_P2_P10_Usuario5=Tasa_Fallas_Cargas(5)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
164 Tasa_Fallas_P1_P2_P10_Usuario6=Tasa_Fallas_Cargas(6)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
165 Tasa_Fallas_P1_P2_P10_Usuario7=Tasa_Fallas_Cargas(7)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
166 Tasa_Fallas_P1_P2_P10_Usuario8=Tasa_Fallas_Cargas(8)+
    Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
```

```
167 Tasa_Fallas_P1_P2_P10_Usuario9=Tasa_Fallas_Cargas(9)+
       Tasa_Fallas_Troncales(1)+sum(Tasa_Fallas_Troncales(5:9));
168 Tasa_Fallas_P1_P2_P10_Usuario10=Tasa_Fallas_Cargas(10)+sum(
       Tasa_Fallas_Troncales);
169 Tasa_Fallas_P1_P2_P10_Usuario11=Tasa_Fallas_Cargas(11)+sum(
       Tasa_Fallas_Troncales);
170 Tasa_Fallas_P1_P2_P10_Usuario12=Tasa_Fallas_Cargas(12)+sum(
       Tasa_Fallas_Troncales);
171 Tasa_Fallas_P1_P2_P10_Usuario13=Tasa_Fallas_Cargas(13)+sum(
       Tasa_Fallas_Troncales(1:9))+sum(Tasa_Fallas_Troncales(13:14));
172 Tasa_Fallas_P1_P2_P10_Usuario14=Tasa_Fallas_Cargas(14)+sum(
       Tasa_Fallas_Troncales(1:9))+sum(Tasa_Fallas_Troncales(13:14));
173
174 Tasa_Fallas_P1_P2_P10_Usuarios=[Tasa_Fallas_P1_P2_P10_Usuario1
       Tasa_Fallas_P1_P2_P10_Usuario2 Tasa_Fallas_P1_P2_P10_Usuario3
       Tasa_Fallas_P1_P2_P10_Usuario4 Tasa_Fallas_P1_P2_P10_Usuario5
       Tasa_Fallas_P1_P2_P10_Usuario6 Tasa_Fallas_P1_P2_P10_Usuario7
       Tasa_Fallas_P1_P2_P10_Usuario8 Tasa_Fallas_P1_P2_P10_Usuario9
       Tasa_Fallas_P1_P2_P10_Usuario10 Tasa_Fallas_P1_P2_P10_Usuario11
       Tasa_Fallas_P1_P2_P10_Usuario12 Tasa_Fallas_P1_P2_P10_Usuario13
       Tasa_Fallas_P1_P2_P10_Usuario14];
175
176 Duracion_Reparacion_P1_P2_P10_Usuario1=U_Cargas(1)+U_Troncales(1)+
       sum(U_Troncales(5:9));
177 Duracion_Reparacion_P1_P2_P10_Usuario2=U_Cargas(2)+sum(U_Troncales
       (1:9))+sum(U_Troncales(13:14));
178 Duracion_Reparacion_P1_P2_P10_Usuario3=U_Cargas(3)+sum(U_Troncales
       (1:9))+sum(U_Troncales(13:14));
179 Duracion_Reparacion_P1_P2_P10_Usuario4=U_Cargas(4)+sum(U_Troncales
       (1:9))+sum(U_Troncales(13:14));
180 Duracion_Reparacion_P1_P2_P10_Usuario5=U_Cargas(5)+U_Troncales(1)+
       sum(U_Troncales(5:9));
181 Duracion_Reparacion_P1_P2_P10_Usuario6=U_Cargas(6)+U_Troncales(1)+
       sum(U_Troncales(5:9));
182 Duracion_Reparacion_P1_P2_P10_Usuario7=U_Cargas(7)+U_Troncales(1)+
       sum(U_Troncales(5:9));
183 Duracion_Reparacion_P1_P2_P10_Usuario8=U_Cargas(8)+U_Troncales(1)+
       sum(U_Troncales(5:9));
184 Duracion_Reparacion_P1_P2_P10_Usuario9=U_Cargas(9)+U_Troncales(1)+
       sum(U_Troncales(5:9));
185 Duracion_Reparacion_P1_P2_P10_Usuario10=U_Cargas(10)+sum(U_Troncales
       );
186 Duracion_Reparacion_P1_P2_P10_Usuario11=U_Cargas(11)+sum(U_Troncales
       );
187 Duracion_Reparacion_P1_P2_P10_Usuario12=U_Cargas(12)+sum(U_Troncales
       );
188 Duracion_Reparacion_P1_P2_P10_Usuario13=U_Cargas(13)+sum(U_Troncales
       (1:9))+sum(U_Troncales(13:14));
189 Duracion_Reparacion_P1_P2_P10_Usuario14=U_Cargas(14)+sum(U_Troncales
       (1:9))+sum(U_Troncales(13:14));
190
191 Duracion_Reparacion_P1_P2_P10_Usuarios=[
```

```
    Duracion_Reparacion_P1_P2_P10_Usuario1
    Duracion_Reparacion_P1_P2_P10_Usuario2
    Duracion_Reparacion_P1_P2_P10_Usuario3
    Duracion_Reparacion_P1_P2_P10_Usuario4
    Duracion_Reparacion_P1_P2_P10_Usuario5
    Duracion_Reparacion_P1_P2_P10_Usuario6
    Duracion_Reparacion_P1_P2_P10_Usuario7
    Duracion_Reparacion_P1_P2_P10_Usuario8
    Duracion_Reparacion_P1_P2_P10_Usuario9
    Duracion_Reparacion_P1_P2_P10_Usuario10
    Duracion_Reparacion_P1_P2_P10_Usuario11
    Duracion_Reparacion_P1_P2_P10_Usuario12
    Duracion_Reparacion_P1_P2_P10_Usuario13
    Duracion_Reparacion_P1_P2_P10_Usuario14];
192
193 SAIFI_P1_P2_P10 = (Tasa_Fallas_P1_P2_P10_Usuario1*Cantidad_Usuarios
    (1)+Tasa_Fallas_P1_P2_P10_Usuario2*Cantidad_Usuarios(2)+
    Tasa_Fallas_P1_P2_P10_Usuario3*Cantidad_Usuarios(3)+
    Tasa_Fallas_P1_P2_P10_Usuario4*Cantidad_Usuarios(4)+
    Tasa_Fallas_P1_P2_P10_Usuario5*Cantidad_Usuarios(5)+
    Tasa_Fallas_P1_P2_P10_Usuario6*Cantidad_Usuarios(6)+
    Tasa_Fallas_P1_P2_P10_Usuario7*Cantidad_Usuarios(7)+
    Tasa_Fallas_P1_P2_P10_Usuario8*Cantidad_Usuarios(8)+
    Tasa_Fallas_P1_P2_P10_Usuario9*Cantidad_Usuarios(9)+
    Tasa_Fallas_P1_P2_P10_Usuario10*Cantidad_Usuarios(10)+
    Tasa_Fallas_P1_P2_P10_Usuario11*Cantidad_Usuarios(11)+
    Tasa_Fallas_P1_P2_P10_Usuario12*Cantidad_Usuarios(12)+
    Tasa_Fallas_P1_P2_P10_Usuario13*Cantidad_Usuarios(13)+
    Tasa_Fallas_P1_P2_P10_Usuario14*Cantidad_Usuarios(14))/
    Total_Consumidores; % Number of average failures per year per
    user
194 SAIDI_P1_P2_P10 = (Duracion_Reparacion_P1_P2_P10_Usuario1*
    Cantidad_Usuarios(1)+Duracion_Reparacion_P1_P2_P10_Usuario2*
    Cantidad_Usuarios(2)+Duracion_Reparacion_P1_P2_P10_Usuario3*
    Cantidad_Usuarios(3)+Duracion_Reparacion_P1_P2_P10_Usuario4*
    Cantidad_Usuarios(4)+Duracion_Reparacion_P1_P2_P10_Usuario5*
    Cantidad_Usuarios(5)+Duracion_Reparacion_P1_P2_P10_Usuario6*
    Cantidad_Usuarios(6)+Duracion_Reparacion_P1_P2_P10_Usuario7*
    Cantidad_Usuarios(7)+Duracion_Reparacion_P1_P2_P10_Usuario8*
    Cantidad_Usuarios(8)+Duracion_Reparacion_P1_P2_P10_Usuario9*
    Cantidad_Usuarios(9)+Duracion_Reparacion_P1_P2_P10_Usuario10*
    Cantidad_Usuarios(10)+Duracion_Reparacion_P1_P2_P10_Usuario11*
    Cantidad_Usuarios(11)+Duracion_Reparacion_P1_P2_P10_Usuario12*
    Cantidad_Usuarios(12)+Duracion_Reparacion_P1_P2_P10_Usuario13*
    Cantidad_Usuarios(13)+Duracion_Reparacion_P1_P2_P10_Usuario14*
    Cantidad_Usuarios(14))/Total_Consumidores; % Average repair time
    per year per user
195 CAIDI_P1_P2_P10 = SAIDI_P1_P2_P10 / SAIFI_P1_P2_P10; % Average
    Duration of an Interruption
196 AENS_P1_P2_P10 = (Duracion_Reparacion_P1_P2_P10_Usuario1*
    Demanda_Usuario(1)+Duracion_Reparacion_P1_P2_P10_Usuario2*
    Demanda_Usuario(2)+Duracion_Reparacion_P1_P2_P10_Usuario3*
```

```
    Demanda_Usuario(3)+Duracion_Reparacion_P1_P2_P10_Usuario4*
    Demanda_Usuario(4)+Duracion_Reparacion_P1_P2_P10_Usuario5*
    Demanda_Usuario(5)+Duracion_Reparacion_P1_P2_P10_Usuario6*
    Demanda_Usuario(6)+Duracion_Reparacion_P1_P2_P10_Usuario7*
    Demanda_Usuario(7)+Duracion_Reparacion_P1_P2_P10_Usuario8*
    Demanda_Usuario(8)+Duracion_Reparacion_P1_P2_P10_Usuario9*
    Demanda_Usuario(9)+Duracion_Reparacion_P1_P2_P10_Usuario10*
    Demanda_Usuario(10)+Duracion_Reparacion_P1_P2_P10_Usuario11*
    Demanda_Usuario(11)+Duracion_Reparacion_P1_P2_P10_Usuario12*
    Demanda_Usuario(12)+Duracion_Reparacion_P1_P2_P10_Usuario13*
    Demanda_Usuario(13)+Duracion_Reparacion_P1_P2_P10_Usuario14*
    Demanda_Usuario(14))/Total_Consumidores; % Average energy NOT
    supplied per year per user (kWh Not Supplied / User)
197
198 % Analysis with reclosers in P1, in P3 and in P5
199 Tasa_Fallas_P1_P3_P5_Usuario1=Tasa_Fallas_Cargas(1)+sum(
    Tasa_Fallas_Troncales(1:2))+sum(Tasa_Fallas_Troncales(8:9))+sum(
    Tasa_Fallas_Troncales(10:12));
200 Tasa_Fallas_P1_P3_P5_Usuario2=Tasa_Fallas_Cargas(2)+sum(
    Tasa_Fallas_Troncales(1:2))+sum(Tasa_Fallas_Troncales(8:9))+sum(
    Tasa_Fallas_Troncales(10:12));
201 Tasa_Fallas_P1_P3_P5_Usuario3=Tasa_Fallas_Cargas(3)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7));
202 Tasa_Fallas_P1_P3_P5_Usuario4=Tasa_Fallas_Cargas(4)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7));
203 Tasa_Fallas_P1_P3_P5_Usuario5=Tasa_Fallas_Cargas(5)+sum(
    Tasa_Fallas_Troncales)-(Tasa_Fallas_Troncales(4))-sum(
    Tasa_Fallas_Troncales(13:14));
204 Tasa_Fallas_P1_P3_P5_Usuario6=Tasa_Fallas_Cargas(6)+sum(
    Tasa_Fallas_Troncales)-(Tasa_Fallas_Troncales(4))-sum(
    Tasa_Fallas_Troncales(13:14));
205 Tasa_Fallas_P1_P3_P5_Usuario7=Tasa_Fallas_Cargas(7)+sum(
    Tasa_Fallas_Troncales)-(Tasa_Fallas_Troncales(4))-sum(
    Tasa_Fallas_Troncales(13:14));
206 Tasa_Fallas_P1_P3_P5_Usuario8=Tasa_Fallas_Cargas(8)+sum(
    Tasa_Fallas_Troncales(1:2))+sum(Tasa_Fallas_Troncales(8:9))+sum(
    Tasa_Fallas_Troncales(10:12));
207 Tasa_Fallas_P1_P3_P5_Usuario9=Tasa_Fallas_Cargas(9)+sum(
    Tasa_Fallas_Troncales(1:2))+sum(Tasa_Fallas_Troncales(8:9))+sum(
    Tasa_Fallas_Troncales(10:12));
208 Tasa_Fallas_P1_P3_P5_Usuario10=Tasa_Fallas_Cargas(10)+sum(
    Tasa_Fallas_Troncales(1:2))+sum(Tasa_Fallas_Troncales(8:9))+sum(
    Tasa_Fallas_Troncales(10:12));
209 Tasa_Fallas_P1_P3_P5_Usuario11=Tasa_Fallas_Cargas(11)+sum(
    Tasa_Fallas_Troncales(1:2))+sum(Tasa_Fallas_Troncales(8:9))+sum(
    Tasa_Fallas_Troncales(10:12));
210 Tasa_Fallas_P1_P3_P5_Usuario12=Tasa_Fallas_Cargas(12)+sum(
    Tasa_Fallas_Troncales(1:2))+sum(Tasa_Fallas_Troncales(8:9))+sum(
    Tasa_Fallas_Troncales(10:12));
211 Tasa_Fallas_P1_P3_P5_Usuario13=Tasa_Fallas_Cargas(13)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7));
212 Tasa_Fallas_P1_P3_P5_Usuario14=Tasa_Fallas_Cargas(14)+sum(
```

```
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7));
213
214 Tasa_Fallas_P1_P3_P5_Usuarios=[Tasa_Fallas_P1_P3_P5_Usuario1
    Tasa_Fallas_P1_P3_P5_Usuario2 Tasa_Fallas_P1_P3_P5_Usuario3
    Tasa_Fallas_P1_P3_P5_Usuario4 Tasa_Fallas_P1_P3_P5_Usuario5
    Tasa_Fallas_P1_P3_P5_Usuario6 Tasa_Fallas_P1_P3_P5_Usuario7
    Tasa_Fallas_P1_P3_P5_Usuario8 Tasa_Fallas_P1_P3_P5_Usuario9
    Tasa_Fallas_P1_P3_P5_Usuario10 Tasa_Fallas_P1_P3_P5_Usuario11
    Tasa_Fallas_P1_P3_P5_Usuario12 Tasa_Fallas_P1_P3_P5_Usuario13
    Tasa_Fallas_P1_P3_P5_Usuario14];
215
216 Duracion_Reparacion_P1_P3_P5_Usuario1=U_Cargas(1)+sum(U_Troncales
    (1:2))+sum(U_Troncales(8:9))+sum(U_Troncales(10:12));
217 Duracion_Reparacion_P1_P3_P5_Usuario2=U_Cargas(2)+sum(U_Troncales
    (1:2))+sum(U_Troncales(8:9))+sum(U_Troncales(10:12));
218 Duracion_Reparacion_P1_P3_P5_Usuario3=U_Cargas(3)+sum(U_Troncales)-
    sum(U_Troncales(5:7));
219 Duracion_Reparacion_P1_P3_P5_Usuario4=U_Cargas(4)+sum(U_Troncales)-
    sum(U_Troncales(5:7));
220 Duracion_Reparacion_P1_P3_P5_Usuario5=U_Cargas(5)+sum(U_Troncales)-(
    U_Troncales(4))-sum(U_Troncales(13:14));
221 Duracion_Reparacion_P1_P3_P5_Usuario6=U_Cargas(6)+sum(U_Troncales)-(
    U_Troncales(4))-sum(U_Troncales(13:14));
222 Duracion_Reparacion_P1_P3_P5_Usuario7=U_Cargas(7)+sum(U_Troncales)-(
    U_Troncales(4))-sum(U_Troncales(13:14));
223 Duracion_Reparacion_P1_P3_P5_Usuario8=U_Cargas(8)+sum(U_Troncales
    (1:2))+sum(U_Troncales(8:9))+sum(U_Troncales(10:12));
224 Duracion_Reparacion_P1_P3_P5_Usuario9=U_Cargas(9)+sum(U_Troncales
    (1:2))+sum(U_Troncales(8:9))+sum(U_Troncales(10:12));
225 Duracion_Reparacion_P1_P3_P5_Usuario10=U_Cargas(10)+sum(U_Troncales
    (1:2))+sum(U_Troncales(8:9))+sum(U_Troncales(10:12));
226 Duracion_Reparacion_P1_P3_P5_Usuario11=U_Cargas(11)+sum(U_Troncales
    (1:2))+sum(U_Troncales(8:9))+sum(U_Troncales(10:12));
227 Duracion_Reparacion_P1_P3_P5_Usuario12=U_Cargas(12)+sum(U_Troncales
    (1:2))+sum(U_Troncales(8:9))+sum(U_Troncales(10:12));
228 Duracion_Reparacion_P1_P3_P5_Usuario13=U_Cargas(13)+sum(U_Troncales)
    -sum(U_Troncales(5:7));
229 Duracion_Reparacion_P1_P3_P5_Usuario14=U_Cargas(14)+sum(U_Troncales)
    -sum(U_Troncales(5:7));
230
231 Duracion_Reparacion_P1_P3_P5_Usuarios=[
    Duracion_Reparacion_P1_P3_P5_Usuario1
    Duracion_Reparacion_P1_P3_P5_Usuario2
    Duracion_Reparacion_P1_P3_P5_Usuario3
    Duracion_Reparacion_P1_P3_P5_Usuario4
    Duracion_Reparacion_P1_P3_P5_Usuario5
    Duracion_Reparacion_P1_P3_P5_Usuario6
    Duracion_Reparacion_P1_P3_P5_Usuario7
    Duracion_Reparacion_P1_P3_P5_Usuario8
    Duracion_Reparacion_P1_P3_P5_Usuario9
    Duracion_Reparacion_P1_P3_P5_Usuario10
    Duracion_Reparacion_P1_P3_P5_Usuario11
```

```
    Duracion_Reparacion_P1_P3_P5_Usuario12
    Duracion_Reparacion_P1_P3_P5_Usuario13
    Duracion_Reparacion_P1_P3_P5_Usuario14];
232
233 SAIFI_P1_P3_P5 = (Tasa_Fallas_P1_P3_P5_Usuario1*Cantidad_Usuarios(1)
    +Tasa_Fallas_P1_P3_P5_Usuario2*Cantidad_Usuarios(2)+
    Tasa_Fallas_P1_P3_P5_Usuario3*Cantidad_Usuarios(3)+
    Tasa_Fallas_P1_P3_P5_Usuario4*Cantidad_Usuarios(4)+
    Tasa_Fallas_P1_P3_P5_Usuario5*Cantidad_Usuarios(5)+
    Tasa_Fallas_P1_P3_P5_Usuario6*Cantidad_Usuarios(6)+
    Tasa_Fallas_P1_P3_P5_Usuario7*Cantidad_Usuarios(7)+
    Tasa_Fallas_P1_P3_P5_Usuario8*Cantidad_Usuarios(8)+
    Tasa_Fallas_P1_P3_P5_Usuario9*Cantidad_Usuarios(9)+
    Tasa_Fallas_P1_P3_P5_Usuario10*Cantidad_Usuarios(10)+
    Tasa_Fallas_P1_P3_P5_Usuario11*Cantidad_Usuarios(11)+
    Tasa_Fallas_P1_P3_P5_Usuario12*Cantidad_Usuarios(12)+
    Tasa_Fallas_P1_P3_P5_Usuario13*Cantidad_Usuarios(13)+
    Tasa_Fallas_P1_P3_P5_Usuario14*Cantidad_Usuarios(14))/
    Total_Consumidores % Number of average failures per year per user
234 SAIDI_P1_P3_P5 = (Duracion_Reparacion_P1_P3_P5_Usuario1*
    Cantidad_Usuarios(1)+Duracion_Reparacion_P1_P3_P5_Usuario2*
    Cantidad_Usuarios(2)+Duracion_Reparacion_P1_P3_P5_Usuario3*
    Cantidad_Usuarios(3)+Duracion_Reparacion_P1_P3_P5_Usuario4*
    Cantidad_Usuarios(4)+Duracion_Reparacion_P1_P3_P5_Usuario5*
    Cantidad_Usuarios(5)+Duracion_Reparacion_P1_P3_P5_Usuario6*
    Cantidad_Usuarios(6)+Duracion_Reparacion_P1_P3_P5_Usuario7*
    Cantidad_Usuarios(7)+Duracion_Reparacion_P1_P3_P5_Usuario8*
    Cantidad_Usuarios(8)+Duracion_Reparacion_P1_P3_P5_Usuario9*
    Cantidad_Usuarios(9)+Duracion_Reparacion_P1_P3_P5_Usuario10*
    Cantidad_Usuarios(10)+Duracion_Reparacion_P1_P3_P5_Usuario11*
    Cantidad_Usuarios(11)+Duracion_Reparacion_P1_P3_P5_Usuario12*
    Cantidad_Usuarios(12)+Duracion_Reparacion_P1_P3_P5_Usuario13*
    Cantidad_Usuarios(13)+Duracion_Reparacion_P1_P3_P5_Usuario14*
    Cantidad_Usuarios(14))/Total_Consumidores % Average repair time
    per year per user
235 CAIDI_P1_P3_P5 = SAIDI_P1_P3_P5 / SAIFI_P1_P3_P5 % Average Duration
    of an Interruption
236 AENS_P1_P3_P5 = (Duracion_Reparacion_P1_P3_P5_Usuario1*
    Demanda_Usuario(1)+Duracion_Reparacion_P1_P3_P5_Usuario2*
    Demanda_Usuario(2)+Duracion_Reparacion_P1_P3_P5_Usuario3*
    Demanda_Usuario(3)+Duracion_Reparacion_P1_P3_P5_Usuario4*
    Demanda_Usuario(4)+Duracion_Reparacion_P1_P3_P5_Usuario5*
    Demanda_Usuario(5)+Duracion_Reparacion_P1_P3_P5_Usuario6*
    Demanda_Usuario(6)+Duracion_Reparacion_P1_P3_P5_Usuario7*
    Demanda_Usuario(7)+Duracion_Reparacion_P1_P3_P5_Usuario8*
    Demanda_Usuario(8)+Duracion_Reparacion_P1_P3_P5_Usuario9*
    Demanda_Usuario(9)+Duracion_Reparacion_P1_P3_P5_Usuario10*
    Demanda_Usuario(10)+Duracion_Reparacion_P1_P3_P5_Usuario11*
    Demanda_Usuario(11)+Duracion_Reparacion_P1_P3_P5_Usuario12*
    Demanda_Usuario(12)+Duracion_Reparacion_P1_P3_P5_Usuario13*
    Demanda_Usuario(13)+Duracion_Reparacion_P1_P3_P5_Usuario14*
    Demanda_Usuario(14))/Total_Consumidores % Average energy NOT
```

```
    supplied per year per user (kWh Not Supplied / User)
237
238
239 % Analysis with reclosers in P1, in P3 and in P8
240 Tasa_Fallas_P1_P3_P8_Usuario1=Tasa_Fallas_Cargas(1)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(8:9));
241 Tasa_Fallas_P1_P3_P8_Usuario2=Tasa_Fallas_Cargas(2)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(8:9));
242 Tasa_Fallas_P1_P3_P8_Usuario3=Tasa_Fallas_Cargas(3)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(8:9));
243 Tasa_Fallas_P1_P3_P8_Usuario4=Tasa_Fallas_Cargas(4)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(8:9));
244 Tasa_Fallas_P1_P3_P8_Usuario5=Tasa_Fallas_Cargas(5)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:9));
245 Tasa_Fallas_P1_P3_P8_Usuario6=Tasa_Fallas_Cargas(6)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:9));
246 Tasa_Fallas_P1_P3_P8_Usuario7=Tasa_Fallas_Cargas(7)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:9));
247 Tasa_Fallas_P1_P3_P8_Usuario8=Tasa_Fallas_Cargas(8)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7));
248 Tasa_Fallas_P1_P3_P8_Usuario9=Tasa_Fallas_Cargas(9)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7));
249 Tasa_Fallas_P1_P3_P8_Usuario10=Tasa_Fallas_Cargas(10)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(8:9));
250 Tasa_Fallas_P1_P3_P8_Usuario11=Tasa_Fallas_Cargas(11)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(8:9));
251 Tasa_Fallas_P1_P3_P8_Usuario12=Tasa_Fallas_Cargas(12)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(8:9));
252 Tasa_Fallas_P1_P3_P8_Usuario13=Tasa_Fallas_Cargas(13)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(8:9));
253 Tasa_Fallas_P1_P3_P8_Usuario14=Tasa_Fallas_Cargas(14)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(8:9));
254
255 Tasa_Fallas_P1_P3_P8_Usuarios=[Tasa_Fallas_P1_P3_P8_Usuario1
    Tasa_Fallas_P1_P3_P8_Usuario2 Tasa_Fallas_P1_P3_P8_Usuario3
    Tasa_Fallas_P1_P3_P8_Usuario4 Tasa_Fallas_P1_P3_P8_Usuario5
    Tasa_Fallas_P1_P3_P8_Usuario6 Tasa_Fallas_P1_P3_P8_Usuario7
    Tasa_Fallas_P1_P3_P8_Usuario8 Tasa_Fallas_P1_P3_P8_Usuario9
    Tasa_Fallas_P1_P3_P8_Usuario10 Tasa_Fallas_P1_P3_P8_Usuario11
    Tasa_Fallas_P1_P3_P8_Usuario12 Tasa_Fallas_P1_P3_P8_Usuario13
    Tasa_Fallas_P1_P3_P8_Usuario14];
256
257 Duracion_Reparacion_P1_P3_P8_Usuario1=U_Cargas(1)+sum(U_Troncales)-
```

```
    sum(U_Troncales(5:9));
Duracion_Reparacion_P1_P3_P8_Usuario2=U_Cargas(2)+sum(U_Troncales)-
    sum(U_Troncales(5:9));
Duracion_Reparacion_P1_P3_P8_Usuario3=U_Cargas(3)+sum(U_Troncales)-
    sum(U_Troncales(5:9));
Duracion_Reparacion_P1_P3_P8_Usuario4=U_Cargas(4)+sum(U_Troncales)-
    sum(U_Troncales(5:9));
Duracion_Reparacion_P1_P3_P8_Usuario5=U_Cargas(5)+sum(U_Troncales)-
    sum(U_Troncales(8:9));
Duracion_Reparacion_P1_P3_P8_Usuario6=U_Cargas(6)+sum(U_Troncales)-
    sum(U_Troncales(8:9));
Duracion_Reparacion_P1_P3_P8_Usuario7=U_Cargas(7)+sum(U_Troncales)-
    sum(U_Troncales(8:9));
Duracion_Reparacion_P1_P3_P8_Usuario8=U_Cargas(8)+sum(U_Troncales)-
    sum(U_Troncales(5:7));
Duracion_Reparacion_P1_P3_P8_Usuario9=U_Cargas(9)+sum(U_Troncales)-
    sum(U_Troncales(5:7));
Duracion_Reparacion_P1_P3_P8_Usuario10=U_Cargas(10)+sum(U_Troncales)
    -sum(U_Troncales(8:9));
Duracion_Reparacion_P1_P3_P8_Usuario11=U_Cargas(11)+sum(U_Troncales)
    -sum(U_Troncales(8:9));
Duracion_Reparacion_P1_P3_P8_Usuario12=U_Cargas(12)+sum(U_Troncales)
    -sum(U_Troncales(8:9));
Duracion_Reparacion_P1_P3_P8_Usuario13=U_Cargas(13)+sum(U_Troncales)
    -sum(U_Troncales(8:9));
Duracion_Reparacion_P1_P3_P8_Usuario14=U_Cargas(14)+sum(U_Troncales)
    -sum(U_Troncales(8:9));

Duracion_Reparacion_P1_P3_P8_Usuarios=[
    Duracion_Reparacion_P1_P3_P8_Usuario1
    Duracion_Reparacion_P1_P3_P8_Usuario2
    Duracion_Reparacion_P1_P3_P8_Usuario3
    Duracion_Reparacion_P1_P3_P8_Usuario4
    Duracion_Reparacion_P1_P3_P8_Usuario5
    Duracion_Reparacion_P1_P3_P8_Usuario6
    Duracion_Reparacion_P1_P3_P8_Usuario7
    Duracion_Reparacion_P1_P3_P8_Usuario8
    Duracion_Reparacion_P1_P3_P8_Usuario9
    Duracion_Reparacion_P1_P3_P8_Usuario10
    Duracion_Reparacion_P1_P3_P8_Usuario11
    Duracion_Reparacion_P1_P3_P8_Usuario12
    Duracion_Reparacion_P1_P3_P8_Usuario13
    Duracion_Reparacion_P1_P3_P8_Usuario14];

SAIFI_P1_P3_P8 = (Tasa_Fallas_P1_P3_P8_Usuario1*Cantidad_Usuarios(1)
    +Tasa_Fallas_P1_P3_P8_Usuario2*Cantidad_Usuarios(2)+
    Tasa_Fallas_P1_P3_P8_Usuario3*Cantidad_Usuarios(3)+
    Tasa_Fallas_P1_P3_P8_Usuario4*Cantidad_Usuarios(4)+
    Tasa_Fallas_P1_P3_P8_Usuario5*Cantidad_Usuarios(5)+
    Tasa_Fallas_P1_P3_P8_Usuario6*Cantidad_Usuarios(6)+
    Tasa_Fallas_P1_P3_P8_Usuario7*Cantidad_Usuarios(7)+
    Tasa_Fallas_P1_P3_P8_Usuario8*Cantidad_Usuarios(8)+
```

```
    Tasa_Fallas_P1_P3_P8_Usuario9*Cantidad_Usuarios(9)+
    Tasa_Fallas_P1_P3_P8_Usuario10*Cantidad_Usuarios(10)+
    Tasa_Fallas_P1_P3_P8_Usuario11*Cantidad_Usuarios(11)+
    Tasa_Fallas_P1_P3_P8_Usuario12*Cantidad_Usuarios(12)+
    Tasa_Fallas_P1_P3_P8_Usuario13*Cantidad_Usuarios(13)+
    Tasa_Fallas_P1_P3_P8_Usuario14*Cantidad_Usuarios(14))/
    Total_Consumidores % Number of average failures per year per user
SAIDI_P1_P3_P8 = (Duracion_Reparacion_P1_P3_P8_Usuario1*
    Cantidad_Usuarios(1)+Duracion_Reparacion_P1_P3_P8_Usuario2*
    Cantidad_Usuarios(2)+Duracion_Reparacion_P1_P3_P8_Usuario3*
    Cantidad_Usuarios(3)+Duracion_Reparacion_P1_P3_P8_Usuario4*
    Cantidad_Usuarios(4)+Duracion_Reparacion_P1_P3_P8_Usuario5*
    Cantidad_Usuarios(5)+Duracion_Reparacion_P1_P3_P8_Usuario6*
    Cantidad_Usuarios(6)+Duracion_Reparacion_P1_P3_P8_Usuario7*
    Cantidad_Usuarios(7)+Duracion_Reparacion_P1_P3_P8_Usuario8*
    Cantidad_Usuarios(8)+Duracion_Reparacion_P1_P3_P8_Usuario9*
    Cantidad_Usuarios(9)+Duracion_Reparacion_P1_P3_P8_Usuario10*
    Cantidad_Usuarios(10)+Duracion_Reparacion_P1_P3_P8_Usuario11*
    Cantidad_Usuarios(11)+Duracion_Reparacion_P1_P3_P8_Usuario12*
    Cantidad_Usuarios(12)+Duracion_Reparacion_P1_P3_P8_Usuario13*
    Cantidad_Usuarios(13)+Duracion_Reparacion_P1_P3_P8_Usuario14*
    Cantidad_Usuarios(14))/Total_Consumidores % Average repair time
    per year per user
CAIDI_P1_P3_P8 = SAIDI_P1_P3_P8 / SAIFI_P1_P3_P8 % Average Duration
    of an Interruption
AENS_P1_P3_P8 = (Duracion_Reparacion_P1_P3_P8_Usuario1*
    Demanda_Usuario(1)+Duracion_Reparacion_P1_P3_P8_Usuario2*
    Demanda_Usuario(2)+Duracion_Reparacion_P1_P3_P8_Usuario3*
    Demanda_Usuario(3)+Duracion_Reparacion_P1_P3_P8_Usuario4*
    Demanda_Usuario(4)+Duracion_Reparacion_P1_P3_P8_Usuario5*
    Demanda_Usuario(5)+Duracion_Reparacion_P1_P3_P8_Usuario6*
    Demanda_Usuario(6)+Duracion_Reparacion_P1_P3_P8_Usuario7*
    Demanda_Usuario(7)+Duracion_Reparacion_P1_P3_P8_Usuario8*
    Demanda_Usuario(8)+Duracion_Reparacion_P1_P3_P8_Usuario9*
    Demanda_Usuario(9)+Duracion_Reparacion_P1_P3_P8_Usuario10*
    Demanda_Usuario(10)+Duracion_Reparacion_P1_P3_P8_Usuario11*
    Demanda_Usuario(11)+Duracion_Reparacion_P1_P3_P8_Usuario12*
    Demanda_Usuario(12)+Duracion_Reparacion_P1_P3_P8_Usuario13*
    Demanda_Usuario(13)+Duracion_Reparacion_P1_P3_P8_Usuario14*
    Demanda_Usuario(14))/Total_Consumidores % Average energy NOT
    supplied per year per user (kWh Not Supplied / User)

% Analysis with reclosers in P1, in P3 and in P10
Tasa_Fallas_P1_P3_P10_Usuario1=Tasa_Fallas_Cargas(1)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(3:4))-sum(
    Tasa_Fallas_Troncales(10:14));
Tasa_Fallas_P1_P3_P10_Usuario2=Tasa_Fallas_Cargas(2)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(3:4))-sum(
    Tasa_Fallas_Troncales(10:14));
Tasa_Fallas_P1_P3_P10_Usuario3=Tasa_Fallas_Cargas(3)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(10:12));
Tasa_Fallas_P1_P3_P10_Usuario4=Tasa_Fallas_Cargas(4)+sum(
```

```
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(10:12));
284 Tasa_Fallas_P1_P3_P10_Usuario5=Tasa_Fallas_Cargas(5)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(3:4))-sum(
    Tasa_Fallas_Troncales(10:14));
285 Tasa_Fallas_P1_P3_P10_Usuario6=Tasa_Fallas_Cargas(6)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(3:4))-sum(
    Tasa_Fallas_Troncales(10:14));
286 Tasa_Fallas_P1_P3_P10_Usuario7=Tasa_Fallas_Cargas(7)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(3:4))-sum(
    Tasa_Fallas_Troncales(10:14));
287 Tasa_Fallas_P1_P3_P10_Usuario8=Tasa_Fallas_Cargas(8)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(3:4))-sum(
    Tasa_Fallas_Troncales(10:14));
288 Tasa_Fallas_P1_P3_P10_Usuario9=Tasa_Fallas_Cargas(9)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(3:4))-sum(
    Tasa_Fallas_Troncales(10:14));
289 Tasa_Fallas_P1_P3_P10_Usuario10=Tasa_Fallas_Cargas(10)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(3:4))-sum(
    Tasa_Fallas_Troncales(13:14));
290 Tasa_Fallas_P1_P3_P10_Usuario11=Tasa_Fallas_Cargas(11)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(3:4))-sum(
    Tasa_Fallas_Troncales(13:14));
291 Tasa_Fallas_P1_P3_P10_Usuario12=Tasa_Fallas_Cargas(12)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(3:4))-sum(
    Tasa_Fallas_Troncales(13:14));
292 Tasa_Fallas_P1_P3_P10_Usuario13=Tasa_Fallas_Cargas(13)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(10:12));
293 Tasa_Fallas_P1_P3_P10_Usuario14=Tasa_Fallas_Cargas(14)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(10:12));
294
295 Tasa_Fallas_P1_P3_P10_Usuarios=[Tasa_Fallas_P1_P3_P10_Usuario1
    Tasa_Fallas_P1_P3_P10_Usuario2 Tasa_Fallas_P1_P3_P10_Usuario3
    Tasa_Fallas_P1_P3_P10_Usuario4 Tasa_Fallas_P1_P3_P10_Usuario5
    Tasa_Fallas_P1_P3_P10_Usuario6 Tasa_Fallas_P1_P3_P10_Usuario7
    Tasa_Fallas_P1_P3_P10_Usuario8 Tasa_Fallas_P1_P3_P10_Usuario9
    Tasa_Fallas_P1_P3_P10_Usuario10 Tasa_Fallas_P1_P3_P10_Usuario11
    Tasa_Fallas_P1_P3_P10_Usuario12 Tasa_Fallas_P1_P3_P10_Usuario13
    Tasa_Fallas_P1_P3_P10_Usuario14];
296
297 Duracion_Reparacion_P1_P3_P10_Usuario1=U_Cargas(1)+sum(U_Troncales)-
    sum(U_Troncales(3:4))-sum(U_Troncales(10:14));
298 Duracion_Reparacion_P1_P3_P10_Usuario2=U_Cargas(2)+sum(U_Troncales)-
    sum(U_Troncales(3:4))-sum(U_Troncales(10:14));
299 Duracion_Reparacion_P1_P3_P10_Usuario3=U_Cargas(3)+sum(U_Troncales)-
    sum(U_Troncales(10:12));
300 Duracion_Reparacion_P1_P3_P10_Usuario4=U_Cargas(4)+sum(U_Troncales)-
    sum(U_Troncales(10:12));
301 Duracion_Reparacion_P1_P3_P10_Usuario5=U_Cargas(5)+sum(U_Troncales)-
    sum(U_Troncales(3:4))-sum(U_Troncales(10:14));
302 Duracion_Reparacion_P1_P3_P10_Usuario6=U_Cargas(6)+sum(U_Troncales)-
    sum(U_Troncales(3:4))-sum(U_Troncales(10:14));
303 Duracion_Reparacion_P1_P3_P10_Usuario7=U_Cargas(7)+sum(U_Troncales)-
```

```
    sum(U_Troncales(3:4))-sum(U_Troncales(10:14));
304 Duracion_Reparacion_P1_P3_P10_Usuario8=U_Cargas(8)+sum(U_Troncales)-
    sum(U_Troncales(3:4))-sum(U_Troncales(10:14));
305 Duracion_Reparacion_P1_P3_P10_Usuario9=U_Cargas(9)+sum(U_Troncales)-
    sum(U_Troncales(3:4))-sum(U_Troncales(10:14));
306 Duracion_Reparacion_P1_P3_P10_Usuario10=U_Cargas(10)+sum(U_Troncales
    )-sum(U_Troncales(3:4))-sum(U_Troncales(13:14));
307 Duracion_Reparacion_P1_P3_P10_Usuario11=U_Cargas(11)+sum(U_Troncales
    )-sum(U_Troncales(3:4))-sum(U_Troncales(13:14));
308 Duracion_Reparacion_P1_P3_P10_Usuario12=U_Cargas(12)+sum(U_Troncales
    )-sum(U_Troncales(3:4))-sum(U_Troncales(13:14));
309 Duracion_Reparacion_P1_P3_P10_Usuario13=U_Cargas(13)+sum(U_Troncales
    )-sum(U_Troncales(10:12));
310 Duracion_Reparacion_P1_P3_P10_Usuario14=U_Cargas(14)+sum(U_Troncales
    )-sum(U_Troncales(10:12));
311
312 Duracion_Reparacion_P1_P3_P10_Usuarios=[
    Duracion_Reparacion_P1_P3_P10_Usuario1
    Duracion_Reparacion_P1_P3_P10_Usuario2
    Duracion_Reparacion_P1_P3_P10_Usuario3
    Duracion_Reparacion_P1_P3_P10_Usuario4
    Duracion_Reparacion_P1_P3_P10_Usuario5
    Duracion_Reparacion_P1_P3_P10_Usuario6
    Duracion_Reparacion_P1_P3_P10_Usuario7
    Duracion_Reparacion_P1_P3_P10_Usuario8
    Duracion_Reparacion_P1_P3_P10_Usuario9
    Duracion_Reparacion_P1_P3_P10_Usuario10
    Duracion_Reparacion_P1_P3_P10_Usuario11
    Duracion_Reparacion_P1_P3_P10_Usuario12
    Duracion_Reparacion_P1_P3_P10_Usuario13
    Duracion_Reparacion_P1_P3_P10_Usuario14];
313
314 SAIFI_P1_P3_P10 = (Tasa_Fallas_P1_P3_P10_Usuario1*Cantidad_Usuarios
    (1)+Tasa_Fallas_P1_P3_P10_Usuario2*Cantidad_Usuarios(2)+
    Tasa_Fallas_P1_P3_P10_Usuario3*Cantidad_Usuarios(3)+
    Tasa_Fallas_P1_P3_P10_Usuario4*Cantidad_Usuarios(4)+
    Tasa_Fallas_P1_P3_P10_Usuario5*Cantidad_Usuarios(5)+
    Tasa_Fallas_P1_P3_P10_Usuario6*Cantidad_Usuarios(6)+
    Tasa_Fallas_P1_P3_P10_Usuario7*Cantidad_Usuarios(7)+
    Tasa_Fallas_P1_P3_P10_Usuario8*Cantidad_Usuarios(8)+
    Tasa_Fallas_P1_P3_P10_Usuario9*Cantidad_Usuarios(9)+
    Tasa_Fallas_P1_P3_P10_Usuario10*Cantidad_Usuarios(10)+
    Tasa_Fallas_P1_P3_P10_Usuario11*Cantidad_Usuarios(11)+
    Tasa_Fallas_P1_P3_P10_Usuario12*Cantidad_Usuarios(12)+
    Tasa_Fallas_P1_P3_P10_Usuario13*Cantidad_Usuarios(13)+
    Tasa_Fallas_P1_P3_P10_Usuario14*Cantidad_Usuarios(14))/
    Total_Consumidores % Number of average failures per year per user
315 SAIDI_P1_P3_P10 = (Duracion_Reparacion_P1_P3_P10_Usuario1*
    Cantidad_Usuarios(1)+Duracion_Reparacion_P1_P3_P10_Usuario2*
    Cantidad_Usuarios(2)+Duracion_Reparacion_P1_P3_P10_Usuario3*
    Cantidad_Usuarios(3)+Duracion_Reparacion_P1_P3_P10_Usuario4*
    Cantidad_Usuarios(4)+Duracion_Reparacion_P1_P3_P10_Usuario5*
```

```
    Cantidad_Usuarios(5)+Duracion_Reparacion_P1_P3_P10_Usuario6*
    Cantidad_Usuarios(6)+Duracion_Reparacion_P1_P3_P10_Usuario7*
    Cantidad_Usuarios(7)+Duracion_Reparacion_P1_P3_P10_Usuario8*
    Cantidad_Usuarios(8)+Duracion_Reparacion_P1_P3_P10_Usuario9*
    Cantidad_Usuarios(9)+Duracion_Reparacion_P1_P3_P10_Usuario10*
    Cantidad_Usuarios(10)+Duracion_Reparacion_P1_P3_P10_Usuario11*
    Cantidad_Usuarios(11)+Duracion_Reparacion_P1_P3_P10_Usuario12*
    Cantidad_Usuarios(12)+Duracion_Reparacion_P1_P3_P10_Usuario13*
    Cantidad_Usuarios(13)+Duracion_Reparacion_P1_P3_P10_Usuario14*
    Cantidad_Usuarios(14))/Total_Consumidores % Average repair time
    per year per user
CAIDI_P1_P3_P10 = SAIDI_P1_P3_P10 / SAIFI_P1_P3_P10 % Average
    Duration of an Interruption
AENS_P1_P3_P10 = (Duracion_Reparacion_P1_P3_P10_Usuario1*
    Demanda_Usuario(1)+Duracion_Reparacion_P1_P3_P10_Usuario2*
    Demanda_Usuario(2)+Duracion_Reparacion_P1_P3_P10_Usuario3*
    Demanda_Usuario(3)+Duracion_Reparacion_P1_P3_P10_Usuario4*
    Demanda_Usuario(4)+Duracion_Reparacion_P1_P3_P10_Usuario5*
    Demanda_Usuario(5)+Duracion_Reparacion_P1_P3_P10_Usuario6*
    Demanda_Usuario(6)+Duracion_Reparacion_P1_P3_P10_Usuario7*
    Demanda_Usuario(7)+Duracion_Reparacion_P1_P3_P10_Usuario8*
    Demanda_Usuario(8)+Duracion_Reparacion_P1_P3_P10_Usuario9*
    Demanda_Usuario(9)+Duracion_Reparacion_P1_P3_P10_Usuario10*
    Demanda_Usuario(10)+Duracion_Reparacion_P1_P3_P10_Usuario11*
    Demanda_Usuario(11)+Duracion_Reparacion_P1_P3_P10_Usuario12*
    Demanda_Usuario(12)+Duracion_Reparacion_P1_P3_P10_Usuario13*
    Demanda_Usuario(13)+Duracion_Reparacion_P1_P3_P10_Usuario14*
    Demanda_Usuario(14))/Total_Consumidores % Average energy NOT
    supplied per year per user (kWh Not Supplied / User)

% Analysis with reclosers in P1, in P5 and in P8
Tasa_Fallas_P1_P5_P8_Usuario1=Tasa_Fallas_Cargas(1)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:9));
Tasa_Fallas_P1_P5_P8_Usuario2=Tasa_Fallas_Cargas(2)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:9));
Tasa_Fallas_P1_P5_P8_Usuario3=Tasa_Fallas_Cargas(3)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:9));
Tasa_Fallas_P1_P5_P8_Usuario4=Tasa_Fallas_Cargas(4)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:9));
Tasa_Fallas_P1_P5_P8_Usuario5=Tasa_Fallas_Cargas(5)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:9));
Tasa_Fallas_P1_P5_P8_Usuario6=Tasa_Fallas_Cargas(6)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:9));
Tasa_Fallas_P1_P5_P8_Usuario7=Tasa_Fallas_Cargas(7)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:9));
Tasa_Fallas_P1_P5_P8_Usuario8=Tasa_Fallas_Cargas(8)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7));
Tasa_Fallas_P1_P5_P8_Usuario9=Tasa_Fallas_Cargas(9)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7));
Tasa_Fallas_P1_P5_P8_Usuario10=Tasa_Fallas_Cargas(10)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:9));
Tasa_Fallas_P1_P5_P8_Usuario11=Tasa_Fallas_Cargas(11)+sum(
```

```
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:9));
331 Tasa_Fallas_P1_P5_P8_Usuario12=Tasa_Fallas_Cargas(12)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:9));
332 Tasa_Fallas_P1_P5_P8_Usuario13=Tasa_Fallas_Cargas(13)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:9));
333 Tasa_Fallas_P1_P5_P8_Usuario14=Tasa_Fallas_Cargas(14)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:9));
334
335 Tasa_Fallas_P1_P5_P8_Usuarios=[Tasa_Fallas_P1_P5_P8_Usuario1
    Tasa_Fallas_P1_P5_P8_Usuario2 Tasa_Fallas_P1_P5_P8_Usuario3
    Tasa_Fallas_P1_P5_P8_Usuario4 Tasa_Fallas_P1_P5_P8_Usuario5
    Tasa_Fallas_P1_P5_P8_Usuario6 Tasa_Fallas_P1_P5_P8_Usuario7
    Tasa_Fallas_P1_P5_P8_Usuario8 Tasa_Fallas_P1_P5_P8_Usuario9
    Tasa_Fallas_P1_P5_P8_Usuario10 Tasa_Fallas_P1_P5_P8_Usuario11
    Tasa_Fallas_P1_P5_P8_Usuario12 Tasa_Fallas_P1_P5_P8_Usuario13
    Tasa_Fallas_P1_P5_P8_Usuario14];
336
337 Duracion_Reparacion_P1_P5_P8_Usuario1=U_Cargas(1)+sum(U_Troncales)-
    sum(U_Troncales(5:9));
338 Duracion_Reparacion_P1_P5_P8_Usuario2=U_Cargas(2)+sum(U_Troncales)-
    sum(U_Troncales(5:9));
339 Duracion_Reparacion_P1_P5_P8_Usuario3=U_Cargas(3)+sum(U_Troncales)-
    sum(U_Troncales(5:9));
340 Duracion_Reparacion_P1_P5_P8_Usuario4=U_Cargas(4)+sum(U_Troncales)-
    sum(U_Troncales(5:9));
341 Duracion_Reparacion_P1_P5_P8_Usuario5=U_Cargas(5)+sum(U_Troncales)-
    sum(U_Troncales(8:9));
342 Duracion_Reparacion_P1_P5_P8_Usuario6=U_Cargas(6)+sum(U_Troncales)-
    sum(U_Troncales(8:9));
343 Duracion_Reparacion_P1_P5_P8_Usuario7=U_Cargas(7)+sum(U_Troncales)-
    sum(U_Troncales(8:9));
344 Duracion_Reparacion_P1_P5_P8_Usuario8=U_Cargas(8)+sum(U_Troncales)-
    sum(U_Troncales(5:7));
345 Duracion_Reparacion_P1_P5_P8_Usuario9=U_Cargas(9)+sum(U_Troncales)-
    sum(U_Troncales(5:7));
346 Duracion_Reparacion_P1_P5_P8_Usuario10=U_Cargas(10)+sum(U_Troncales)
    -sum(U_Troncales(5:9));
347 Duracion_Reparacion_P1_P5_P8_Usuario11=U_Cargas(11)+sum(U_Troncales)
    -sum(U_Troncales(5:9));
348 Duracion_Reparacion_P1_P5_P8_Usuario12=U_Cargas(12)+sum(U_Troncales)
    -sum(U_Troncales(5:9));
349 Duracion_Reparacion_P1_P5_P8_Usuario13=U_Cargas(13)+sum(U_Troncales)
    -sum(U_Troncales(5:9));
350 Duracion_Reparacion_P1_P5_P8_Usuario14=U_Cargas(14)+sum(U_Troncales)
    -sum(U_Troncales(5:9));
351
352 Duracion_Reparacion_P1_P5_P8_Usuarios=[
    Duracion_Reparacion_P1_P5_P8_Usuario1
    Duracion_Reparacion_P1_P5_P8_Usuario2
    Duracion_Reparacion_P1_P5_P8_Usuario3
    Duracion_Reparacion_P1_P5_P8_Usuario4
    Duracion_Reparacion_P1_P5_P8_Usuario5
```

```
    Duracion_Reparacion_P1_P5_P8_Usuario6
    Duracion_Reparacion_P1_P5_P8_Usuario7
    Duracion_Reparacion_P1_P5_P8_Usuario8
    Duracion_Reparacion_P1_P5_P8_Usuario9
    Duracion_Reparacion_P1_P5_P8_Usuario10
    Duracion_Reparacion_P1_P5_P8_Usuario11
    Duracion_Reparacion_P1_P5_P8_Usuario12
    Duracion_Reparacion_P1_P5_P8_Usuario13
    Duracion_Reparacion_P1_P5_P8_Usuario14];
353
354 SAIFI_P1_P5_P8 = (Tasa_Fallas_P1_P5_P8_Usuario1*Cantidad_Usuarios(1)
    +Tasa_Fallas_P1_P5_P8_Usuario2*Cantidad_Usuarios(2)+
    Tasa_Fallas_P1_P5_P8_Usuario3*Cantidad_Usuarios(3)+
    Tasa_Fallas_P1_P5_P8_Usuario4*Cantidad_Usuarios(4)+
    Tasa_Fallas_P1_P5_P8_Usuario5*Cantidad_Usuarios(5)+
    Tasa_Fallas_P1_P5_P8_Usuario6*Cantidad_Usuarios(6)+
    Tasa_Fallas_P1_P5_P8_Usuario7*Cantidad_Usuarios(7)+
    Tasa_Fallas_P1_P5_P8_Usuario8*Cantidad_Usuarios(8)+
    Tasa_Fallas_P1_P5_P8_Usuario9*Cantidad_Usuarios(9)+
    Tasa_Fallas_P1_P5_P8_Usuario10*Cantidad_Usuarios(10)+
    Tasa_Fallas_P1_P5_P8_Usuario11*Cantidad_Usuarios(11)+
    Tasa_Fallas_P1_P5_P8_Usuario12*Cantidad_Usuarios(12)+
    Tasa_Fallas_P1_P5_P8_Usuario13*Cantidad_Usuarios(13)+
    Tasa_Fallas_P1_P5_P8_Usuario14*Cantidad_Usuarios(14))/
    Total_Consumidores % Number of average failures per year per user
355 SAIDI_P1_P5_P8 = (Duracion_Reparacion_P1_P5_P8_Usuario1*
    Cantidad_Usuarios(1)+Duracion_Reparacion_P1_P5_P8_Usuario2*
    Cantidad_Usuarios(2)+Duracion_Reparacion_P1_P5_P8_Usuario3*
    Cantidad_Usuarios(3)+Duracion_Reparacion_P1_P5_P8_Usuario4*
    Cantidad_Usuarios(4)+Duracion_Reparacion_P1_P5_P8_Usuario5*
    Cantidad_Usuarios(5)+Duracion_Reparacion_P1_P5_P8_Usuario6*
    Cantidad_Usuarios(6)+Duracion_Reparacion_P1_P5_P8_Usuario7*
    Cantidad_Usuarios(7)+Duracion_Reparacion_P1_P5_P8_Usuario8*
    Cantidad_Usuarios(8)+Duracion_Reparacion_P1_P5_P8_Usuario9*
    Cantidad_Usuarios(9)+Duracion_Reparacion_P1_P5_P8_Usuario10*
    Cantidad_Usuarios(10)+Duracion_Reparacion_P1_P5_P8_Usuario11*
    Cantidad_Usuarios(11)+Duracion_Reparacion_P1_P5_P8_Usuario12*
    Cantidad_Usuarios(12)+Duracion_Reparacion_P1_P5_P8_Usuario13*
    Cantidad_Usuarios(13)+Duracion_Reparacion_P1_P5_P8_Usuario14*
    Cantidad_Usuarios(14))/Total_Consumidores % Average repair time
    per year per user
356 CAIDI_P1_P5_P8 = SAIDI_P1_P5_P8 / SAIFI_P1_P5_P8 % Average Duration
    of an Interruption
357 AENS_P1_P5_P8 = (Duracion_Reparacion_P1_P5_P8_Usuario1*
    Demanda_Usuario(1)+Duracion_Reparacion_P1_P5_P8_Usuario2*
    Demanda_Usuario(2)+Duracion_Reparacion_P1_P5_P8_Usuario3*
    Demanda_Usuario(3)+Duracion_Reparacion_P1_P5_P8_Usuario4*
    Demanda_Usuario(4)+Duracion_Reparacion_P1_P5_P8_Usuario5*
    Demanda_Usuario(5)+Duracion_Reparacion_P1_P5_P8_Usuario6*
    Demanda_Usuario(6)+Duracion_Reparacion_P1_P5_P8_Usuario7*
    Demanda_Usuario(7)+Duracion_Reparacion_P1_P5_P8_Usuario8*
    Demanda_Usuario(8)+Duracion_Reparacion_P1_P5_P8_Usuario9*
```

```
    Demanda_Usuario(9)+Duracion_Reparacion_P1_P5_P8_Usuario10*
    Demanda_Usuario(10)+Duracion_Reparacion_P1_P5_P8_Usuario11*
    Demanda_Usuario(11)+Duracion_Reparacion_P1_P5_P8_Usuario12*
    Demanda_Usuario(12)+Duracion_Reparacion_P1_P5_P8_Usuario13*
    Demanda_Usuario(13)+Duracion_Reparacion_P1_P5_P8_Usuario14*
    Demanda_Usuario(14))/Total_Consumidores % Average energy NOT
    supplied per year per user (kWh Not Supplied / User)
358
359 % Analysis with reclosers in P1, in P5 and in P10
360 Tasa_Fallas_P1_P5_P10_Usuario1=Tasa_Fallas_Cargas(1)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(10:12));
361 Tasa_Fallas_P1_P5_P10_Usuario2=Tasa_Fallas_Cargas(2)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(10:12));
362 Tasa_Fallas_P1_P5_P10_Usuario3=Tasa_Fallas_Cargas(3)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(10:12));
363 Tasa_Fallas_P1_P5_P10_Usuario4=Tasa_Fallas_Cargas(4)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(10:12));
364 Tasa_Fallas_P1_P5_P10_Usuario5=Tasa_Fallas_Cargas(5)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(10:12));
365 Tasa_Fallas_P1_P5_P10_Usuario6=Tasa_Fallas_Cargas(6)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(10:12));
366 Tasa_Fallas_P1_P5_P10_Usuario7=Tasa_Fallas_Cargas(7)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(10:12));
367 Tasa_Fallas_P1_P5_P10_Usuario8=Tasa_Fallas_Cargas(8)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(10:12));
368 Tasa_Fallas_P1_P5_P10_Usuario9=Tasa_Fallas_Cargas(9)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(10:12));
369 Tasa_Fallas_P1_P5_P10_Usuario10=Tasa_Fallas_Cargas(10)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7));
370 Tasa_Fallas_P1_P5_P10_Usuario11=Tasa_Fallas_Cargas(11)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7));
371 Tasa_Fallas_P1_P5_P10_Usuario12=Tasa_Fallas_Cargas(12)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7));
372 Tasa_Fallas_P1_P5_P10_Usuario13=Tasa_Fallas_Cargas(13)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(10:12));
373 Tasa_Fallas_P1_P5_P10_Usuario14=Tasa_Fallas_Cargas(14)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(5:7))-sum(
    Tasa_Fallas_Troncales(10:12));
374
375 Tasa_Fallas_P1_P5_P10_Usuarios=[Tasa_Fallas_P1_P5_P10_Usuario1
    Tasa_Fallas_P1_P5_P10_Usuario2 Tasa_Fallas_P1_P5_P10_Usuario3
    Tasa_Fallas_P1_P5_P10_Usuario4 Tasa_Fallas_P1_P5_P10_Usuario5
    Tasa_Fallas_P1_P5_P10_Usuario6 Tasa_Fallas_P1_P5_P10_Usuario7
    Tasa_Fallas_P1_P5_P10_Usuario8 Tasa_Fallas_P1_P5_P10_Usuario9
    Tasa_Fallas_P1_P5_P10_Usuario10 Tasa_Fallas_P1_P5_P10_Usuario11
```

```
    Tasa_Fallas_P1_P5_P10_Usuario12 Tasa_Fallas_P1_P5_P10_Usuario13
    Tasa_Fallas_P1_P5_P10_Usuario14];
376
377 Duracion_Reparacion_P1_P5_P10_Usuario1=U_Cargas(1)+sum(U_Troncales)-
    sum(U_Troncales(5:7))-sum(U_Troncales(10:12));
378 Duracion_Reparacion_P1_P5_P10_Usuario2=U_Cargas(2)+sum(U_Troncales)-
    sum(U_Troncales(5:7))-sum(U_Troncales(10:12));
379 Duracion_Reparacion_P1_P5_P10_Usuario3=U_Cargas(3)+sum(U_Troncales)-
    sum(U_Troncales(5:7))-sum(U_Troncales(10:12));
380 Duracion_Reparacion_P1_P5_P10_Usuario4=U_Cargas(4)+sum(U_Troncales)-
    sum(U_Troncales(5:7))-sum(U_Troncales(10:12));
381 Duracion_Reparacion_P1_P5_P10_Usuario5=U_Cargas(5)+sum(U_Troncales)-
    sum(U_Troncales(10:12));
382 Duracion_Reparacion_P1_P5_P10_Usuario6=U_Cargas(6)+sum(U_Troncales)-
    sum(U_Troncales(10:12));
383 Duracion_Reparacion_P1_P5_P10_Usuario7=U_Cargas(7)+sum(U_Troncales)-
    sum(U_Troncales(10:12));
384 Duracion_Reparacion_P1_P5_P10_Usuario8=U_Cargas(8)+sum(U_Troncales)-
    sum(U_Troncales(5:7))-sum(U_Troncales(10:12));
385 Duracion_Reparacion_P1_P5_P10_Usuario9=U_Cargas(9)+sum(U_Troncales)-
    sum(U_Troncales(5:7))-sum(U_Troncales(10:12));
386 Duracion_Reparacion_P1_P5_P10_Usuario10=U_Cargas(10)+sum(U_Troncales
    )-sum(U_Troncales(5:7));
387 Duracion_Reparacion_P1_P5_P10_Usuario11=U_Cargas(11)+sum(U_Troncales
    )-sum(U_Troncales(5:7));
388 Duracion_Reparacion_P1_P5_P10_Usuario12=U_Cargas(12)+sum(U_Troncales
    )-sum(U_Troncales(5:7));
389 Duracion_Reparacion_P1_P5_P10_Usuario13=U_Cargas(13)+sum(U_Troncales
    )-sum(U_Troncales(5:7))-sum(U_Troncales(10:12));
390 Duracion_Reparacion_P1_P5_P10_Usuario14=U_Cargas(14)+sum(U_Troncales
    )-sum(U_Troncales(5:7))-sum(U_Troncales(10:12));
391
392 Duracion_Reparacion_P1_P5_P10_Usuarios=[
    Duracion_Reparacion_P1_P5_P10_Usuario1
    Duracion_Reparacion_P1_P5_P10_Usuario2
    Duracion_Reparacion_P1_P5_P10_Usuario3
    Duracion_Reparacion_P1_P5_P10_Usuario4
    Duracion_Reparacion_P1_P5_P10_Usuario5
    Duracion_Reparacion_P1_P5_P10_Usuario6
    Duracion_Reparacion_P1_P5_P10_Usuario7
    Duracion_Reparacion_P1_P5_P10_Usuario8
    Duracion_Reparacion_P1_P5_P10_Usuario9
    Duracion_Reparacion_P1_P5_P10_Usuario10
    Duracion_Reparacion_P1_P5_P10_Usuario11
    Duracion_Reparacion_P1_P5_P10_Usuario12
    Duracion_Reparacion_P1_P5_P10_Usuario13
    Duracion_Reparacion_P1_P5_P10_Usuario14];
393
394 SAIFI_P1_P5_P10 = (Tasa_Fallas_P1_P5_P10_Usuario1*Cantidad_Usuarios
    (1)+Tasa_Fallas_P1_P5_P10_Usuario2*Cantidad_Usuarios(2)+
    Tasa_Fallas_P1_P5_P10_Usuario3*Cantidad_Usuarios(3)+
    Tasa_Fallas_P1_P5_P10_Usuario4*Cantidad_Usuarios(4)+
```

```
    Tasa_Fallas_P1_P5_P10_Usuario5*Cantidad_Usuarios(5)+
    Tasa_Fallas_P1_P5_P10_Usuario6*Cantidad_Usuarios(6)+
    Tasa_Fallas_P1_P5_P10_Usuario7*Cantidad_Usuarios(7)+
    Tasa_Fallas_P1_P5_P10_Usuario8*Cantidad_Usuarios(8)+
    Tasa_Fallas_P1_P5_P10_Usuario9*Cantidad_Usuarios(9)+
    Tasa_Fallas_P1_P5_P10_Usuario10*Cantidad_Usuarios(10)+
    Tasa_Fallas_P1_P5_P10_Usuario11*Cantidad_Usuarios(11)+
    Tasa_Fallas_P1_P5_P10_Usuario12*Cantidad_Usuarios(12)+
    Tasa_Fallas_P1_P5_P10_Usuario13*Cantidad_Usuarios(13)+
    Tasa_Fallas_P1_P5_P10_Usuario14*Cantidad_Usuarios(14))/
    Total_Consumidores % Number of average failures per year per user
SAIDI_P1_P5_P10 = (Duracion_Reparacion_P1_P5_P10_Usuario1*
    Cantidad_Usuarios(1)+Duracion_Reparacion_P1_P5_P10_Usuario2*
    Cantidad_Usuarios(2)+Duracion_Reparacion_P1_P5_P10_Usuario3*
    Cantidad_Usuarios(3)+Duracion_Reparacion_P1_P5_P10_Usuario4*
    Cantidad_Usuarios(4)+Duracion_Reparacion_P1_P5_P10_Usuario5*
    Cantidad_Usuarios(5)+Duracion_Reparacion_P1_P5_P10_Usuario6*
    Cantidad_Usuarios(6)+Duracion_Reparacion_P1_P5_P10_Usuario7*
    Cantidad_Usuarios(7)+Duracion_Reparacion_P1_P5_P10_Usuario8*
    Cantidad_Usuarios(8)+Duracion_Reparacion_P1_P5_P10_Usuario9*
    Cantidad_Usuarios(9)+Duracion_Reparacion_P1_P5_P10_Usuario10*
    Cantidad_Usuarios(10)+Duracion_Reparacion_P1_P5_P10_Usuario11*
    Cantidad_Usuarios(11)+Duracion_Reparacion_P1_P5_P10_Usuario12*
    Cantidad_Usuarios(12)+Duracion_Reparacion_P1_P5_P10_Usuario13*
    Cantidad_Usuarios(13)+Duracion_Reparacion_P1_P5_P10_Usuario14*
    Cantidad_Usuarios(14))/Total_Consumidores % Average repair time
    per year per user
CAIDI_P1_P5_P10 = SAIDI_P1_P5_P10 / SAIFI_P1_P5_P10 % Average
    Duration of an Interruption
AENS_P1_P5_P10 = (Duracion_Reparacion_P1_P5_P10_Usuario1*
    Demanda_Usuario(1)+Duracion_Reparacion_P1_P5_P10_Usuario2*
    Demanda_Usuario(2)+Duracion_Reparacion_P1_P5_P10_Usuario3*
    Demanda_Usuario(3)+Duracion_Reparacion_P1_P5_P10_Usuario4*
    Demanda_Usuario(4)+Duracion_Reparacion_P1_P5_P10_Usuario5*
    Demanda_Usuario(5)+Duracion_Reparacion_P1_P5_P10_Usuario6*
    Demanda_Usuario(6)+Duracion_Reparacion_P1_P5_P10_Usuario7*
    Demanda_Usuario(7)+Duracion_Reparacion_P1_P5_P10_Usuario8*
    Demanda_Usuario(8)+Duracion_Reparacion_P1_P5_P10_Usuario9*
    Demanda_Usuario(9)+Duracion_Reparacion_P1_P5_P10_Usuario10*
    Demanda_Usuario(10)+Duracion_Reparacion_P1_P5_P10_Usuario11*
    Demanda_Usuario(11)+Duracion_Reparacion_P1_P5_P10_Usuario12*
    Demanda_Usuario(12)+Duracion_Reparacion_P1_P5_P10_Usuario13*
    Demanda_Usuario(13)+Duracion_Reparacion_P1_P5_P10_Usuario14*
    Demanda_Usuario(14))/Total_Consumidores % Average energy NOT
    supplied per year per user (kWh Not Supplied / User)

% Analysis with reclosers in P1, in P8 and in P10
Tasa_Fallas_P1_P8_P10_Usuario1=Tasa_Fallas_Cargas(1)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:12));
Tasa_Fallas_P1_P8_P10_Usuario2=Tasa_Fallas_Cargas(2)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:12));
Tasa_Fallas_P1_P8_P10_Usuario3=Tasa_Fallas_Cargas(3)+sum(
```

```
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:12));
403 Tasa_Fallas_P1_P8_P10_Usuario4=Tasa_Fallas_Cargas(4)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:12));
404 Tasa_Fallas_P1_P8_P10_Usuario5=Tasa_Fallas_Cargas(5)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:12));
405 Tasa_Fallas_P1_P8_P10_Usuario6=Tasa_Fallas_Cargas(6)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:12));
406 Tasa_Fallas_P1_P8_P10_Usuario7=Tasa_Fallas_Cargas(7)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:12));
407 Tasa_Fallas_P1_P8_P10_Usuario8=Tasa_Fallas_Cargas(8)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(10:12));
408 Tasa_Fallas_P1_P8_P10_Usuario9=Tasa_Fallas_Cargas(9)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(10:12));
409 Tasa_Fallas_P1_P8_P10_Usuario10=Tasa_Fallas_Cargas(10)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:9));
410 Tasa_Fallas_P1_P8_P10_Usuario11=Tasa_Fallas_Cargas(11)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:9));
411 Tasa_Fallas_P1_P8_P10_Usuario12=Tasa_Fallas_Cargas(12)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:9));
412 Tasa_Fallas_P1_P8_P10_Usuario13=Tasa_Fallas_Cargas(13)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:12));
413 Tasa_Fallas_P1_P8_P10_Usuario14=Tasa_Fallas_Cargas(14)+sum(
    Tasa_Fallas_Troncales)-sum(Tasa_Fallas_Troncales(8:12));
414
415 Tasa_Fallas_P1_P8_P10_Usuarios=[Tasa_Fallas_P1_P8_P10_Usuario1
    Tasa_Fallas_P1_P8_P10_Usuario2 Tasa_Fallas_P1_P8_P10_Usuario3
    Tasa_Fallas_P1_P8_P10_Usuario4 Tasa_Fallas_P1_P8_P10_Usuario5
    Tasa_Fallas_P1_P8_P10_Usuario6 Tasa_Fallas_P1_P8_P10_Usuario7
    Tasa_Fallas_P1_P8_P10_Usuario8 Tasa_Fallas_P1_P8_P10_Usuario9
    Tasa_Fallas_P1_P8_P10_Usuario10 Tasa_Fallas_P1_P8_P10_Usuario11
    Tasa_Fallas_P1_P8_P10_Usuario12 Tasa_Fallas_P1_P8_P10_Usuario13
    Tasa_Fallas_P1_P8_P10_Usuario14];
416
417 Duracion_Reparacion_P1_P8_P10_Usuario1=U_Cargas(1)+sum(U_Troncales)-
    sum(U_Troncales(8:12));
418 Duracion_Reparacion_P1_P8_P10_Usuario2=U_Cargas(2)+sum(U_Troncales)-
    sum(U_Troncales(8:12));
419 Duracion_Reparacion_P1_P8_P10_Usuario3=U_Cargas(3)+sum(U_Troncales)-
    sum(U_Troncales(8:12));
420 Duracion_Reparacion_P1_P8_P10_Usuario4=U_Cargas(4)+sum(U_Troncales)-
    sum(U_Troncales(8:12));
421 Duracion_Reparacion_P1_P8_P10_Usuario5=U_Cargas(5)+sum(U_Troncales)-
    sum(U_Troncales(8:12));
422 Duracion_Reparacion_P1_P8_P10_Usuario6=U_Cargas(6)+sum(U_Troncales)-
    sum(U_Troncales(8:12));
423 Duracion_Reparacion_P1_P8_P10_Usuario7=U_Cargas(7)+sum(U_Troncales)-
    sum(U_Troncales(8:12));
424 Duracion_Reparacion_P1_P8_P10_Usuario8=U_Cargas(8)+sum(U_Troncales)-
    sum(U_Troncales(10:12));
425 Duracion_Reparacion_P1_P8_P10_Usuario9=U_Cargas(9)+sum(U_Troncales)-
    sum(U_Troncales(10:12));
426 Duracion_Reparacion_P1_P8_P10_Usuario10=U_Cargas(10)+sum(U_Troncales
```

```
    )-sum(U_Troncales(8:9));
427 Duracion_Reparacion_P1_P8_P10_Usuario11=U_Cargas(11)+sum(U_Troncales
    )-sum(U_Troncales(8:9));
428 Duracion_Reparacion_P1_P8_P10_Usuario12=U_Cargas(12)+sum(U_Troncales
    )-sum(U_Troncales(8:9));
429 Duracion_Reparacion_P1_P8_P10_Usuario13=U_Cargas(13)+sum(U_Troncales
    )-sum(U_Troncales(8:12));
430 Duracion_Reparacion_P1_P8_P10_Usuario14=U_Cargas(14)+sum(U_Troncales
    )-sum(U_Troncales(8:12));
431
432 Duracion_Reparacion_P1_P8_P10_Usuarios=[
    Duracion_Reparacion_P1_P8_P10_Usuario1
    Duracion_Reparacion_P1_P8_P10_Usuario2
    Duracion_Reparacion_P1_P8_P10_Usuario3
    Duracion_Reparacion_P1_P8_P10_Usuario4
    Duracion_Reparacion_P1_P8_P10_Usuario5
    Duracion_Reparacion_P1_P8_P10_Usuario6
    Duracion_Reparacion_P1_P8_P10_Usuario7
    Duracion_Reparacion_P1_P8_P10_Usuario8
    Duracion_Reparacion_P1_P8_P10_Usuario9
    Duracion_Reparacion_P1_P8_P10_Usuario10
    Duracion_Reparacion_P1_P8_P10_Usuario11
    Duracion_Reparacion_P1_P8_P10_Usuario12
    Duracion_Reparacion_P1_P8_P10_Usuario13
    Duracion_Reparacion_P1_P8_P10_Usuario14];
433
434 SAIFI_P1_P8_P10 = (Tasa_Fallas_P1_P8_P10_Usuario1*Cantidad_Usuarios
    (1)+Tasa_Fallas_P1_P8_P10_Usuario2*Cantidad_Usuarios(2)+
    Tasa_Fallas_P1_P8_P10_Usuario3*Cantidad_Usuarios(3)+
    Tasa_Fallas_P1_P8_P10_Usuario4*Cantidad_Usuarios(4)+
    Tasa_Fallas_P1_P8_P10_Usuario5*Cantidad_Usuarios(5)+
    Tasa_Fallas_P1_P8_P10_Usuario6*Cantidad_Usuarios(6)+
    Tasa_Fallas_P1_P8_P10_Usuario7*Cantidad_Usuarios(7)+
    Tasa_Fallas_P1_P8_P10_Usuario8*Cantidad_Usuarios(8)+
    Tasa_Fallas_P1_P8_P10_Usuario9*Cantidad_Usuarios(9)+
    Tasa_Fallas_P1_P8_P10_Usuario10*Cantidad_Usuarios(10)+
    Tasa_Fallas_P1_P8_P10_Usuario11*Cantidad_Usuarios(11)+
    Tasa_Fallas_P1_P8_P10_Usuario12*Cantidad_Usuarios(12)+
    Tasa_Fallas_P1_P8_P10_Usuario13*Cantidad_Usuarios(13)+
    Tasa_Fallas_P1_P8_P10_Usuario14*Cantidad_Usuarios(14))/
    Total_Consumidores % Number of average failures per year per user
435 SAIDI_P1_P8_P10 = (Duracion_Reparacion_P1_P8_P10_Usuario1*
    Cantidad_Usuarios(1)+Duracion_Reparacion_P1_P8_P10_Usuario2*
    Cantidad_Usuarios(2)+Duracion_Reparacion_P1_P8_P10_Usuario3*
    Cantidad_Usuarios(3)+Duracion_Reparacion_P1_P8_P10_Usuario4*
    Cantidad_Usuarios(4)+Duracion_Reparacion_P1_P8_P10_Usuario5*
    Cantidad_Usuarios(5)+Duracion_Reparacion_P1_P8_P10_Usuario6*
    Cantidad_Usuarios(6)+Duracion_Reparacion_P1_P8_P10_Usuario7*
    Cantidad_Usuarios(7)+Duracion_Reparacion_P1_P8_P10_Usuario8*
    Cantidad_Usuarios(8)+Duracion_Reparacion_P1_P8_P10_Usuario9*
    Cantidad_Usuarios(9)+Duracion_Reparacion_P1_P8_P10_Usuario10*
    Cantidad_Usuarios(10)+Duracion_Reparacion_P1_P8_P10_Usuario11*
```

```
    Cantidad_Usuarios(11)+Duracion_Reparacion_P1_P8_P10_Usuario12*
    Cantidad_Usuarios(12)+Duracion_Reparacion_P1_P8_P10_Usuario13*
    Cantidad_Usuarios(13)+Duracion_Reparacion_P1_P8_P10_Usuario14*
    Cantidad_Usuarios(14))/Total_Consumidores % Average repair time
    per year per user
CAIDI_P1_P8_P10 = SAIDI_P1_P8_P10 / SAIFI_P1_P8_P10 % Average
    Duration of an Interruption
AENS_P1_P8_P10 = (Duracion_Reparacion_P1_P8_P10_Usuario1*
    Demanda_Usuario(1)+Duracion_Reparacion_P1_P8_P10_Usuario2*
    Demanda_Usuario(2)+Duracion_Reparacion_P1_P8_P10_Usuario3*
    Demanda_Usuario(3)+Duracion_Reparacion_P1_P8_P10_Usuario4*
    Demanda_Usuario(4)+Duracion_Reparacion_P1_P8_P10_Usuario5*
    Demanda_Usuario(5)+Duracion_Reparacion_P1_P8_P10_Usuario6*
    Demanda_Usuario(6)+Duracion_Reparacion_P1_P8_P10_Usuario7*
    Demanda_Usuario(7)+Duracion_Reparacion_P1_P8_P10_Usuario8*
    Demanda_Usuario(8)+Duracion_Reparacion_P1_P8_P10_Usuario9*
    Demanda_Usuario(9)+Duracion_Reparacion_P1_P8_P10_Usuario10*
    Demanda_Usuario(10)+Duracion_Reparacion_P1_P8_P10_Usuario11*
    Demanda_Usuario(11)+Duracion_Reparacion_P1_P8_P10_Usuario12*
    Demanda_Usuario(12)+Duracion_Reparacion_P1_P8_P10_Usuario13*
    Demanda_Usuario(13)+Duracion_Reparacion_P1_P8_P10_Usuario14*
    Demanda_Usuario(14))/Total_Consumidores % Average energy NOT
    supplied per year per user (kWh Not Supplied / User)

% Multicriteria analysis for one (1) additional recloser to P1 and
    P2. They are considering the location in P2 as a previously
    analysed winning alternative. One recloser is considered in P1
    and another in P2, and a third recloser's optimal location is
    calculated as sectionalising coordination.

Matriz_Decision1=[SAIFI_P1_P2_P3 SAIFI_P1_P2_P5 SAIFI_P1_P2_P8
    SAIFI_P1_P2_P10 SAIFI_P1_P3_P5 SAIFI_P1_P3_P8 SAIFI_P1_P3_P10
    SAIFI_P1_P5_P8 SAIFI_P1_P5_P10 SAIFI_P1_P8_P10; ...
     SAIDI_P1_P2_P3 SAIDI_P1_P2_P5 SAIDI_P1_P2_P8 SAIDI_P1_P2_P10
        SAIDI_P1_P3_P5 SAIDI_P1_P3_P8 SAIDI_P1_P3_P10 SAIDI_P1_P5_P8
        SAIDI_P1_P5_P10 SAIDI_P1_P8_P10; ...
     CAIDI_P1_P2_P3 CAIDI_P1_P2_P5 CAIDI_P1_P2_P8 CAIDI_P1_P2_P10
        CAIDI_P1_P3_P5 CAIDI_P1_P3_P8 CAIDI_P1_P3_P10 CAIDI_P1_P5_P8
        CAIDI_P1_P5_P10 CAIDI_P1_P8_P10; ...
     AENS_P1_P2_P3 AENS_P1_P2_P5 AENS_P1_P2_P8 AENS_P1_P2_P10
        AENS_P1_P3_P5 AENS_P1_P3_P8 AENS_P1_P3_P10 AENS_P1_P5_P8
        AENS_P1_P5_P10 AENS_P1_P8_P10]

coders1 = {'P1, P2, P3' , 'P1, P2, P5', 'P1, P2, P8', 'P1, P2, P10',
     'P1, P3, P5', 'P1, P3, P8', 'P1, P3, P10', 'P1, P5, P8', 'P1, P5
    , P10', 'P1, P8, P10'}

for i=1:1:length(Matriz_Decision1(1,:))
     for j=1:1:length(Matriz_Decision1(:,1))
Matriz_Normalizada1(j,:)=(1/sum(Matriz_Decision1(j,:)))*
    Matriz_Decision1(j,:);
Matriz_Normalizada_Rango1(j,i)=(Matriz_Decision1(j,i)-min(
```

```
    Matriz_Decision1(j,:)))/(max(Matriz_Decision1(j,:))-min(
    Matriz_Decision1(j,:))); % Normalization by Range
452      end
453  end
454  Matriz_Normalizada_Rango1;
455  for n=1:1:length(Matriz_Decision1(:,1))
456  Desviacion_estandar1(n,:)=std(Matriz_Normalizada1(n,:));
457  end
458  Desviacion_estandar1;
459  R = corrcoef(Matriz_Normalizada1');
460  for m=1:1:length(Matriz_Decision1(:,1))
461  Ponderacion1=Desviacion_estandar1.*sum(1-R(:,m));
462  end
463  Ponderacion1;
464
465  Ponderacion_Normalizada1=(1/sum(Ponderacion1))*Ponderacion1;
466
467  cont1=0;
468  for k=1:1:j
469      Sumas_Ponderadas1 = Matriz_Normalizada_Rango1(k,:).*
        Ponderacion_Normalizada1 (k) + cont1;
470      cont=Sumas_Ponderadas1;
471  end
472  Sumas_Ponderadas1;
473
474  win_case1 = min (Sumas_Ponderadas1); [Escenario_Ganador1(x)] = find(
    Sumas_Ponderadas1 == win_case1); % position of the winning
    alternative
475  end
476  % Analysis in Montecarlo
477  P1_P2_P3=0;
478  P1_P2_P5=0;
479  P1_P2_P8=0;
480  P1_P2_P10=0;
481  P1_P3_P5=0;
482  P1_P3_P8=0;
483  P1_P3_P10=0;
484  P1_P5_P8=0;
485  P1_P5_P10=0;
486  P1_P8_P10=0;
487
488  for i=1:1:N
489  if Escenario_Ganador1(i)==1
490      P1_P2_P3=1+P1_P2_P3;
491  end
492  if Escenario_Ganador1(i)==2
493      P1_P2_P5=1+P1_P2_P5;
494  end
495  if Escenario_Ganador1(i)==3
496      P1_P2_P8=1+P1_P2_P8;
497  end
498  if Escenario_Ganador1(i)==4
```

```
499      P1_P2_P10=1+P1_P2_P10;
500  end
501  if Escenario_Ganador1(i)==5
502      P1_P3_P5=1+P1_P3_P5;
503  end
504  if Escenario_Ganador1(i)==6
505      P1_P3_P8=1+P1_P3_P8;
506  end
507  if Escenario_Ganador1(i)==7
508      P1_P3_P10=1+P1_P3_P10;
509  end
510  if Escenario_Ganador1(i)==8
511      P1_P5_P8=1+P1_P5_P8;
512  end
513  if Escenario_Ganador1(i)==9
514      P1_P5_P10=1+P1_P5_P10;
515  end
516  if Escenario_Ganador1(i)==10
517      P1_P8_P10=1+P1_P8_P10;
518  end
519  end
520  P1_P2_P3
521  P1_P2_P5
522  P1_P2_P8
523  P1_P2_P10
524  P1_P3_P5
525  P1_P3_P8
526  P1_P3_P10
527  P1_P5_P8
528  P1_P5_P10
529  P1_P8_P10
530  Probabilidad_P1_P2_P3=P1_P2_P3/N
531  Probabilidad_P1_P2_P5=P1_P2_P5/N
532  Probabilidad_P1_P2_P8=P1_P2_P8/N
533  Probabilidad_P1_P2_P10=P1_P2_P10/N
534  Probabilidad_P1_P3_P5=P1_P3_P5/N
535  Probabilidad_P1_P3_P8=P1_P3_P8/N
536  Probabilidad_P1_P3_P10=P1_P3_P10/N
537  Probabilidad_P1_P5_P8=P1_P5_P8/N
538  Probabilidad_P1_P5_P10=P1_P5_P10/N
539  Probabilidad_P1_P8_P10=P1_P8_P10/N
540
541  Suma_Probabilidades=Probabilidad_P1_P2_P3+Probabilidad_P1_P2_P5+
        Probabilidad_P1_P2_P8+Probabilidad_P1_P2_P10+
        Probabilidad_P1_P3_P5+Probabilidad_P1_P3_P8+
        Probabilidad_P1_P3_P10+Probabilidad_P1_P5_P8+
        Probabilidad_P1_P5_P10+Probabilidad_P1_P8_P10
542  toc
543
544  figure(1)
545  pie3([P1_P2_P3 P1_P2_P5 P1_P2_P8 P1_P2_P10 P1_P3_P5 P1_P3_P8
        P1_P3_P10 P1_P5_P8 P1_P5_P10 P1_P8_P10])
```

```
546 legend('P1 P2 P3','P1 P2 P5','P1 P2 P8','P1 P2 P10','P1_P3_P5','
       P1_P3_P8','P1_P3_P10','P1_P5_P8','P1_P5_P10','P1_P8_P10')
547 title('Odds of Winning Montecarlo Alternatives')
548 f = gcf;
549 exportgraphics(f,'Figure53.png','Resolution',800)
550 legend("Position",[0.68938,0.041108,0.29637,0.26344])
```

At the end of this algorithm, a Montecarlo analysis is performed to evaluate the frequency with which each alternative emerges as the winner based on the proposed probabilistic scenarios. Additionally, the algorithm includes a timer, although the computational time used is unimportant since this is a planning problem. The presented algorithm provides all the required results for this analysis and the corresponding figure generated for the study.

3.5 Conclusions

In conclusion, using multicriteria decision methods for solving optimal location problems in electric distribution systems proves crucial. These methods provide a systematic approach to evaluate and compare multiple criteria simultaneously, allowing for a comprehensive analysis of various factors involved in the decision-making process. By considering reliability indicators, cost considerations, constraints on reliability, efficiency and quality of electrical power, and other relevant factors, the multicriteria decision method helps identify the optimal locations for placing reclosers or other system components. This approach enables efficient and adequate decision-making in enhancing the reliability and performance of electric distribution systems.

The proposed method enables a more efficient study of the winning alternative, significantly reducing the computational time as it focuses on exploring only necessary alternatives. However, it is essential to note that the new solution depends on a previously identified dominant solution. The optimization analysis using multicriteria decision is performed across different pseudo-random data generations, providing a more precise understanding of the probability of each switching possibility becoming a winning alternative for the recloser installation. This analysis incorporates the robust probabilistic theory of the Monte Carlo method, which is well-suited for this type of study. The winning probability of each alternative was calculated through 100,000 Monte Carlo scenarios with an absolute error of 0.0032. The alternative of placing two reclosers at P2 and P5 and the existing one at P1 emerged as the winner in 58 per cent of the evaluated alternatives. The proposed method of section coordination was validated through an exhaustive search, demonstrating that it does not compromise the optimal result and consistently identifies the same winning alternative in 64 percent of the random scenarios examined.

It is worth noting that the proposed model is equally applicable to a distribution system where accurate data is available. The optimal switching location and coordination can be precisely determined using the multicriteria method based on the provided data.

In this work, the optimization problem with multicriteria decisions has been formulated. For the compensation problem, the algorithm considers two demand scenarios. It calculates power flows with discretized compensation using an Exhaustive Search algorithm, which determines the location of discretized capacitances among a set of candidate buses. Based on these preliminary results, a second algorithm is developed that assigns weights to each variable, tailored to a minimization criterion. As a result, each compensation scenario obtains a weighted solution, with some solutions discarded based on the dominance criterion. Ultimately, the optimal solution that satisfies the criteria is chosen. This methodology ensures that the compensation solution aligns with correct operational outcomes for reactive power compensation in distribution systems, considering distributed resources.

Furthermore, the research analyzes a case where Total Harmonic Distortion (THD) is not considered an objective criterion. The results demonstrate the conflicts between the analyzed variables in the system under study, which justifies and supports the proposed methodology.

The proposed methodology offers innovative solutions, emphasizing the importance of treating the problem of reactive power compensation as a multicriteria problem and considering a more comprehensive range of quality and efficiency variables than previously addressed by other authors.

Bibliography

[1] A. Aguila Téllez, "Optimización Multicriterio de Flujos de Potencia Reactiva en Sistemas Eléctricos de Distribución", *Tesis de Doctorado*, vol. 2021-06-15, 105 páginas, 2021. DOI: http://hdl.handle.net/20.500.11912/8699. [Online]. Available: https://repository.upb.edu.co/handle/20.500.11912/8699.

[2] A. Águila Téllez, G. López, I. Isaac, and J. González, "Optimal reactive power compensation in electrical distribution systems with distributed resources. Review", *Heliyon*, vol. 4, no. 8, e00746, 2018, ISSN: 24058440. DOI: 10.1016/j.heliyon.2018.e00746. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2405844018332766.

[3] A. Águila, L. Ortiz, R. Orizondo, and G. López, "Optimal location and dimensioning of capacitors in microgrids using a multicriteria decision algorithm", *Heliyon*, vol. 7, no. 9, e08061, 2021, ISSN: 24058440. DOI: 10.1016/j.heliyon.2021.e08061.

[4] A. Aguila Téllez, L. Ortiz, M. Ruiz, K. Narayanan, and S. Varela, "Optimal location of reclosers in electrical distribution systems considering multicriteria decision through the generation of scenarios using the montecarlo method.", *IEEE Access*, pp. 1–1, 2023. DOI: 10.1109/ACCESS.2023.3252411.

[5] G. Niu, M. Wu, Y. Ji, L. Kou, and H. Zhang, "Power supply reliability evaluation method for dc distribution network with power energy router", *China International Conference on Electricity Distribution, CICED*, vol. 2021-April, no. 202005310000033, pp. 138–142, 2021, ISSN: 2161749X. DOI: 10.1109/CICED50259.2021.9556697.

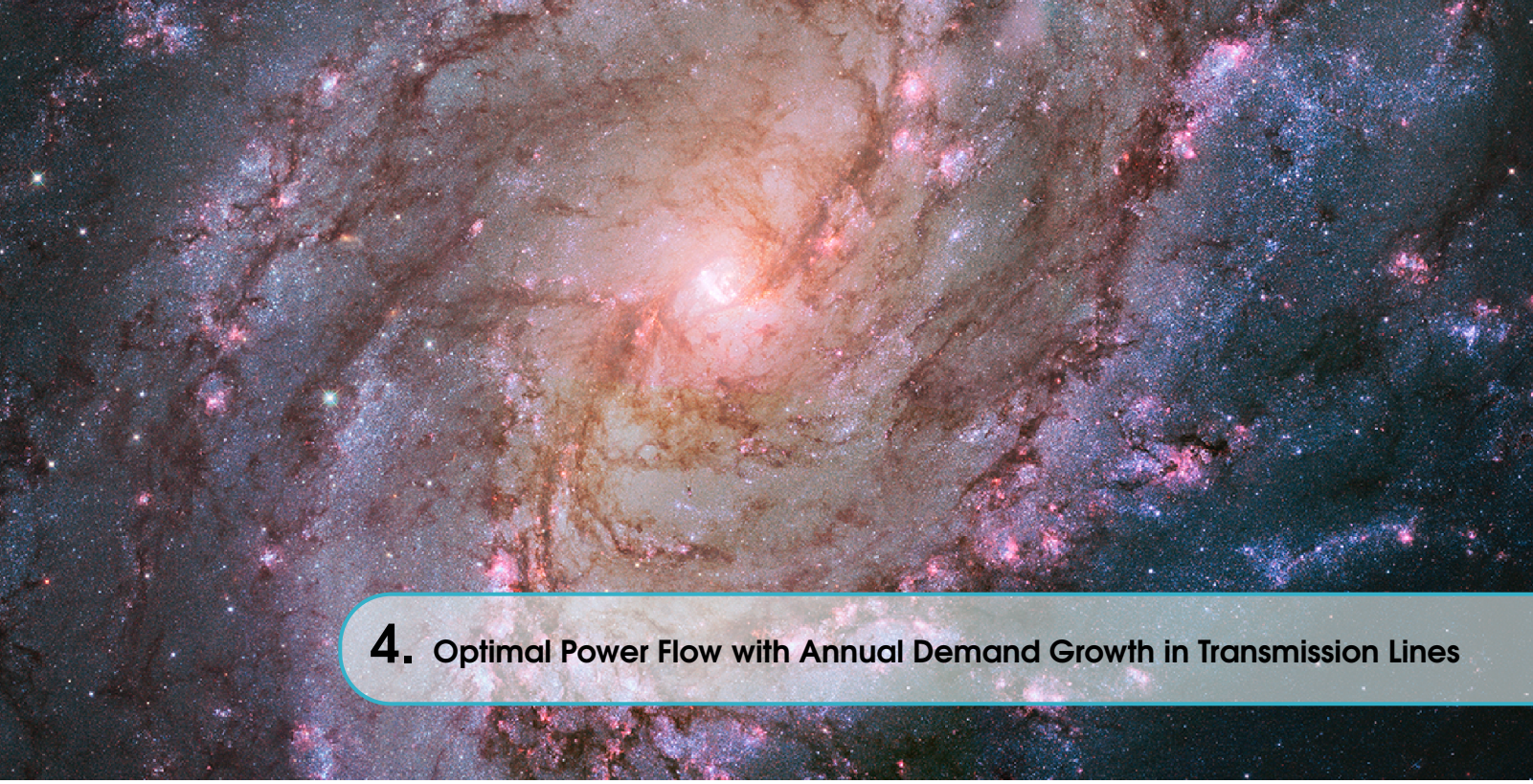
[6] E. Guanochanga, A. Águila, and L. Ortiz, "Multicriteria analysis for optimal reconfiguration of a distribution network in case of failures", *Heliyon*, vol. 9, no. 3, e13723, Mar. 2023, ISSN: 2405-8440. DOI: 10.1016/J.HELIYON.2023.E13723. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2405844023009301.

[7] M. Z. Habib, M. T. Hoq, S. Duvnjak Zarkovic, and N. Taylor, "Impact of the fault location methods on SAIDI of a resonant-earthed distribution system", *2020 IEEE International Conference on Power Systems Technology, POWERCON 2020*, 2020. DOI: 10.1109/POWERCON48463.2020.9230614.

[8] A. Banerjee, S. Chattopadhyay, M. Gavrilas, and G. Grigoras, "Optimization and estimation of reliability indices and cost of Power Distribution System of an urban area by a noble fuzzy-hybrid algorithm", *Applied Soft Computing*, vol. 102, p. 107 078, 2021, ISSN: 15684946. DOI: 10.1016/j.asoc.2021.107078. [Online]. Available: https://doi.org/10.1016/j.asoc.2021.107078.

[9] L. Ortiz, R. Orizondo, A. Aguila, J. W. Gonz, I. Isaac, and J. L. Gabriel, "Hybrid AC / DC microgrid test system simulation : grid-connected mode", *Heliyon*, vol. 5, no. August, p. 21, 2019. DOI: 10.1016/j.heliyon.2019.e02862. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405844019365211.

[10] L. Ortiz, R. Orizondo, A. Aguila, I. Isaac, and J. Gabriel, "Hybrid ac/dc microgrid (hmg) test system simulation (https://www. mathworks. com/matlabcentral/fileexchange/73878-hybrid-ac-dc-microgrid-hmg-test-systemsimulation)", *MATLAB Central File Exchange*, 2020.



4. Optimal Power Flow with Annual Demand Growth in Transmission Lines

Ph.D. Diego Carrión Galarza is Professor of the Master's Degree Program in Electricity at Universidad Politécnica Salesiana (Ecuador)
Email: dcarrion@ups.edu.ec
https://orcid.org/0000-0002-0704-700X
DOI: 10.17163/abyaups.44.349

4.1 Introduction

Electrical power systems (EPS) are very complex systems due to the large number of elements involved in their operation; they have a large number of generation plants of different technologies, meshed connection systems at high voltage levels, bidirectional distribution systems, and end users with greater demands and iteration with the electricity market. With all this, the network topology of the electrical system is not so simple to analyze, which is why it is necessary to have new strategies for its control (1)–(6).

With the background above, new concepts have been created based on economic aspects, becoming an optimization problem, calling it now an optimal power flow problem (OPF). Since then, several resolution techniques have been developed (7).

Optimization plays an important role when planning the operation and managing to analyze the dynamism of the electricity market; therefore, the search for the optimal solution for the operation and management of energy systems has become a necessity, considering the high costs of different technologies, that energy resources are limited and the growing demand for electricity energy (8).

The objective of the OPF is to find the ideal configuration for optimal operation while minimizing operating costs and maximizing reliability. For an OPF, constraints involving operating costs, system losses, voltage level, transmission line capacity, equipment operating limits, and load demand are considered (9), (10).

According to the selected objective function and constraints, different mathematical formulations exist to solve an OPF. They can be classified as follows:

- Linear problems in which the objectives and constraints are given in linear forms with continuous control variables.
- Nonlinear problems, where the objectives and constraints; or both combined are nonlinear with continuous control variables.
- Nonlinear mixed integer problems with discrete and continuous control variables.

Instead, if analyzed from the point of view of the main attributes, they can be divided as follows:

- Lambda iteration method, in which the losses can be represented by a matrix (B), or the penalty factors can be calculated separately by a power flow. It forms the basis of many standard online economic dispatch programs.
- Gradient method is a slow convergence method and is difficult to solve in the presence of inequality constraints.
- Newton's method, whose convergence is very fast, can give problems with inequality constraints.
- Linear programming method, which is a method that can work with inequality constraints, functions, and nonlinear objective constraints handled by linearization.
- Interior point method: Another fully developed and widely used method for OPF. It easily handles inequality constraints.

In (11), the authors have used the optimal AC power flow model (OPF-AC) to study the behavior with high penetration of renewable energies by comparing the model with others and verifying its accuracy. In contrast, (12) shows the optimal DC power flow model (OPF-DC) applied to the analysis of power systems with the centralized and decentralized generation, which details an approximation of the EPS behavior. The OPF-AC problem for OPF calculation based on the gradient and Newton methods consists of finding the active and reactive power values, the voltage magnitudes of any generator unit to minimize the operating cost, and the fulfillment of various constraints. While OPF-DC is applied to calculate the load flow in the model, it is characterized by ignoring the losses, and the focus is only on the real power. This solution is usually used as the OPF-AC approximation to something more straightforward as a linear circuit analysis problem. Although an optimal AC power flow is a more accurate calculation of the steady-state

behavior of the EPS (13), correlation parameters can be obscured due to the complexity and nonlinearity of its characteristic equations. On the other hand, OPF-DC methods and their software requirements are simple; their models can be optimized efficiently, and the required minimum network data need little execution time (14). The optimal DC power flow approximation has been widely used in literature for nodal pricing and bottleneck analysis across transmission lines (15).

Another technique in which reactive power and EPS losses are not neglected to find an approximate solution to the optimal power flows is the linearization of the optimal AC power flows (LOPF-AC), for which numerical methods are used to obtain an approximation of the variables that have a power different from unity (16).

Due to the mathematical complexity of the optimization models that allow finding the solution to the OPF, the use of specialized software is necessary. This chapter deals with the help of Matlab and GAMS programs for modeling and optimization, respectively, but for this purpose, an interface has been created that makes GAMS a Matlab tool. With it, using the solution to the power flows, any other study can be performed, such as N-1 contingency analysis, reliability studies, and even stability studies.

4.2 Configuration

To configure the Matlab – GAMS interface, the first thing to do is to configure GAMS to allow its variables to belong to the computer's environment variables; for this purpose, when installing GAMS, the advanced installation mode option must be selected, see figure 4.1, so that after selecting the installation folder and other options, the environment variables can be enabled, see figure 4.2.

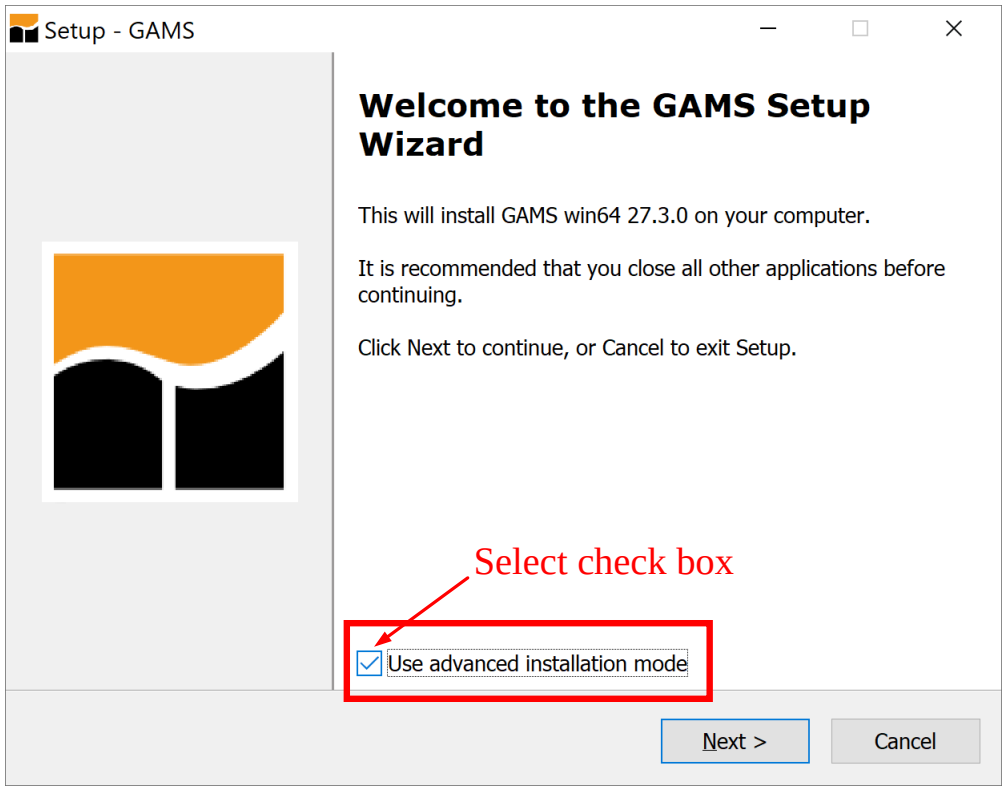

Setup - GAMS
Welcome to the GAMS Setup Wizard
This will install GAMS win64 27.3.0 on your computer.
It is recommended that you close all other applications before continuing.
Click Next to continue, or Cancel to exit Setup.
Select check box
Use advanced installation mode
Next >
Cancel

Figure 4.1: Initialization of GAMS installation process

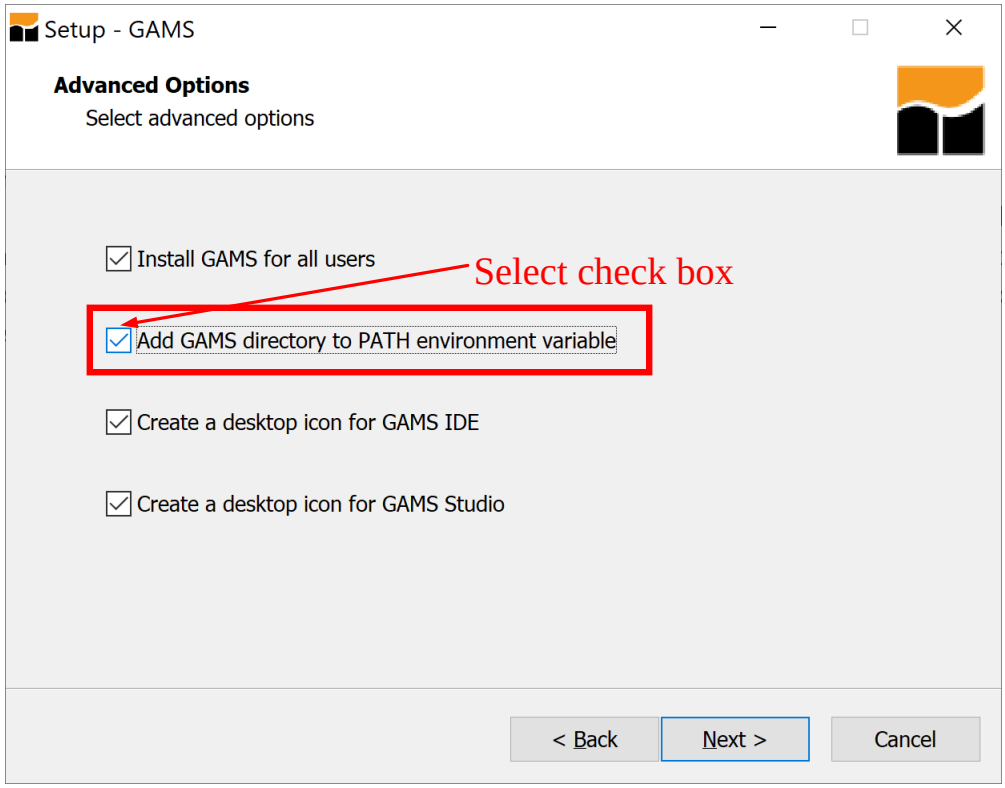

Setup - GAMS
Advanced Options
Select advanced options
Install GAMS for all users
Select check box
Add GAMS directory to PATH environment variable
Create a desktop icon for GAMS IDE
Create a desktop icon for GAMS Studio
< Back
Next >
Cancel

Figure 4.2: Enable GAMS path environment variable

Once GAMS is installed with this procedure, the environment variables of the computer must be configured, for which in Windows Explorer it must be made right click on "This PC" and from the menu select "Properties," there a system configuration window will be displayed, and in the right side it must be chosen "Advanced system settings," see figure 4.3; after that in the window that is displayed the option of environment variables must be selected, see figure 4.4. There it will allow configuring the environment variables for the user and those of the whole system; it is recommended to make the configuration for the two territories; for which the environment variable path must be edited and there to place the location where GAMS was installed, see figure 4.5.

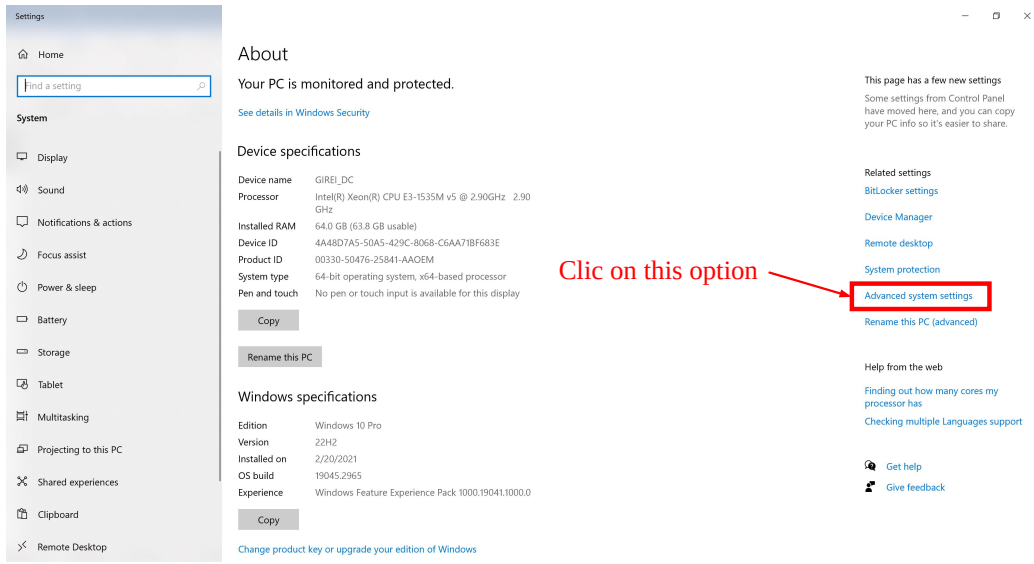


Figure 4.3: Configuration advanced system settings

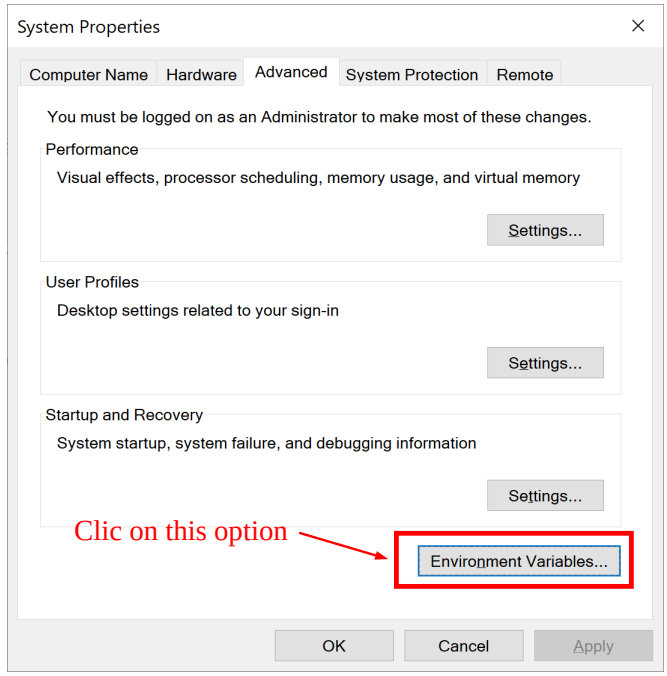


Figure 4.4: Choose environment variables

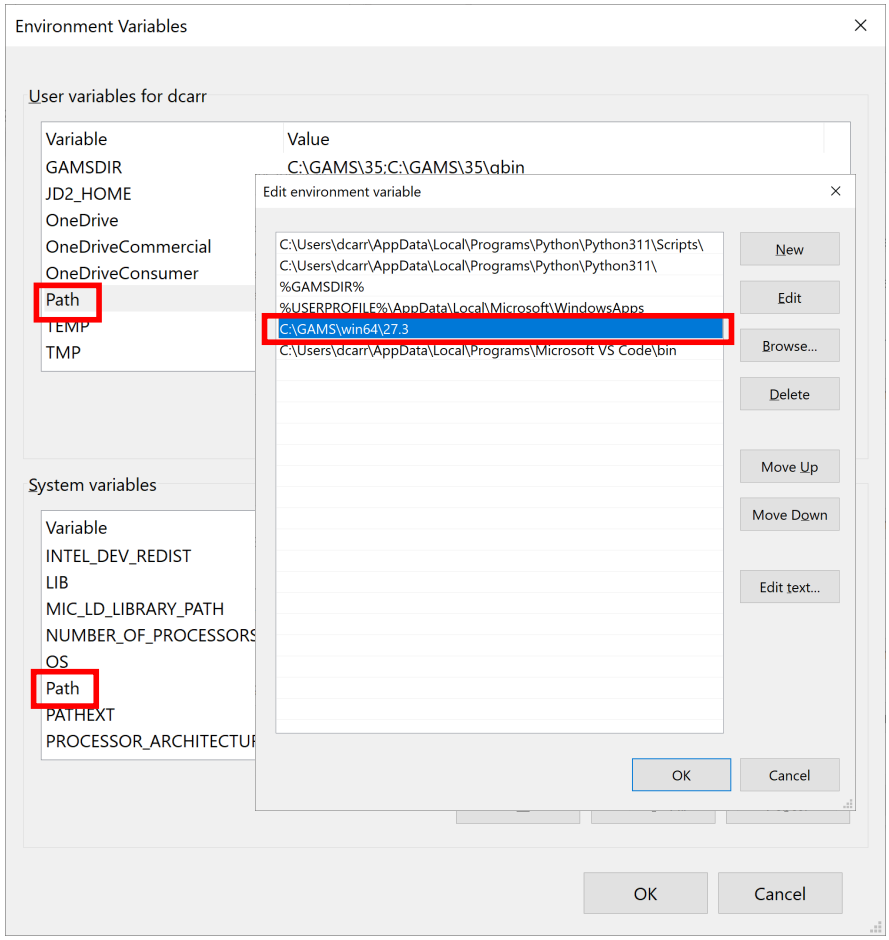


Figure 4.5: Set GAMS path

Once this process is completed, the environment variables must be configured in Matlab, for which the "Set Path" button must be selected in the toolbar of the "Home menu," see figure 4.6, and there is a similar process to the Windows configuration performed, a folder is added, and this will correspond to the location where GAMS was installed, see figure 4.7. Now you must verify that the process has been done correctly by writing the following code in the command window:

```
1 system 'gams ? lo=3'
```

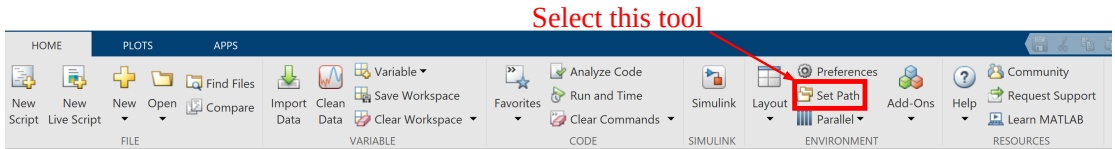


Figure 4.6: Set GAMS path

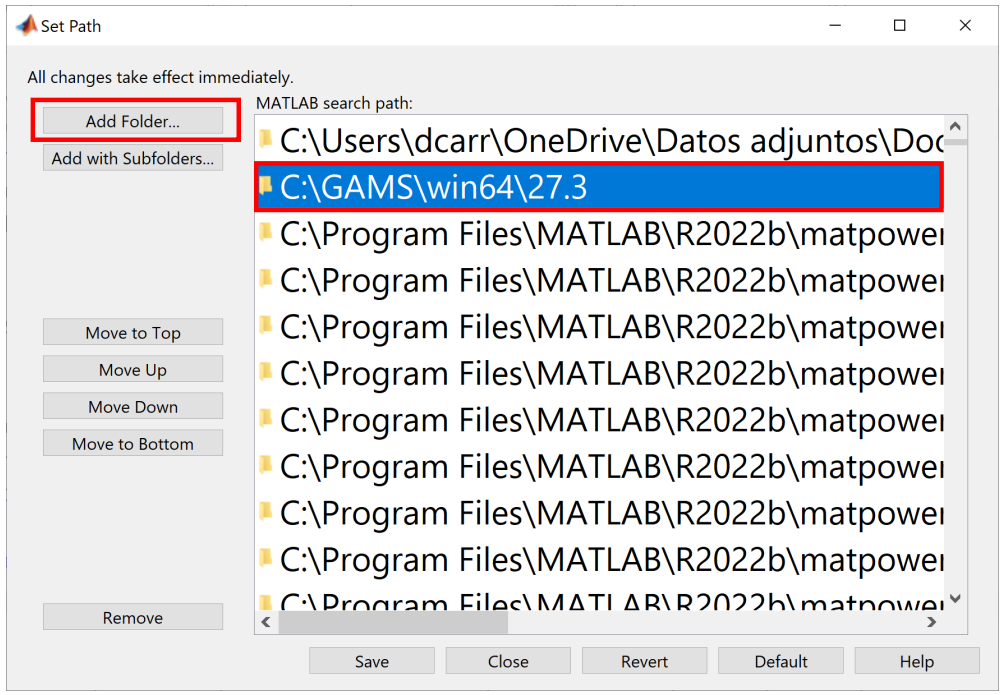


Figure 4.7: Configure GAMS path in Matlab

There you will see that the demo license information is reflected, execution time, and ans=0, see figure 4.8; if ans is equal to another value, it is recommended to review the configuration process because there may be an error and the environment variables have not been configured correctly.

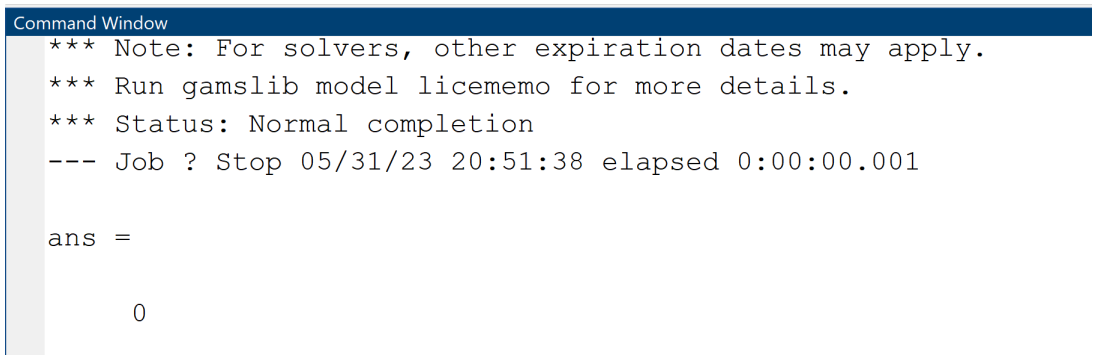

Command Window
*** Note: For solvers, other expiration dates may apply.
*** Run gamslib model licememo for more details.
*** Status: Normal completion
--- Job ? Stop 05/31/23 20:51:38 elapsed 0:00:00.001
ans =
0

Figure 4.8: Configure GAMS path in Matlab

4.3 Optimal power flows

Power flows are a technique that allows quantifying each of the electrical variables in the operation of power systems, for which different styles such as Gauss – Seidel,

Newton − Raphson, and optimal power flows have been developed. The first two methods respond to iterative techniques, and the shortest convergence time of the two is Newton − Raphson; on the other hand, the optimal power flows respond to numerical optimization techniques that seek to minimize operating costs and maximize the technical resources of the power system.

Table 4.1: Variables related to the optimization model

Symbol	Description
i, j	Nodes index
C_g	Generation cost
P_g	Generator active power
Q_g	Generator reactive power
V_i	Nodes voltage
δ_i, δ_j	Nodes voltage angle
V_i^{max}	Upper voltage limit
V_i^{min}	Lower voltage limit
δ_i^{max}	Upper angular limit
δ_i^{min}	Lower angular limit
P_{i-j}	Active power flow
Q_{i-j}	Reactive power flow
S_{i-j}	Apparent power flow
SIL	Surge impedance load
B_{i-j}	Line susceptance
G_{i-j}	Line conductance
P_D	Demand active power
Q_D	Demand reactive power

4.3.1 DC Optimal power flow

The OPF-DC focuses on analyzing the EPS discarding reactive power, losses in the system, and that the magnitude of the voltage at each node is one pu; an OPF-DC seeks to minimize costs, which is similar to maximizing the welfare of society by obtaining a quality service at a lower cost. The objective function of the OPF-DC is defined in (4.1), which seeks to minimize production costs, which is subject to (4.2) that conditions the characteristics of the generation plants, (4.3) determines the maximum power flow that can circulate through the transmission lines, (4.4) the power balance between generation, demand, and power flow; (4.5) that determines the power flow that circulates through a transmission line depending on its susceptance; and (4.6) is the voltage angle limits (17)–(19).

$ObjetiveFunction$:

$$min\ Z = \sum_{g=1}^{N_G} C_g P_g \tag{4.1}$$

$Subjectto:$

$$P_{G_i}^{min} \leq P_{G_i} \leq P_{G_i}^{max} \tag{4.2}$$

$$-SIL \leq P_{i-j} \leq SIL \tag{4.3}$$

$$\sum_{i=1}^{n_{bus}} \sum_{j=1}^{n_{bus}} P_{i-j} = P_{G_i} - P_{D_i};\ \ \forall\, i,j\ \in\ N_B \tag{4.4}$$

$$P_{i-j} = B_{i-j}\,(\delta_i - \delta_j) \tag{4.5}$$

$$\delta_i^{min} \leq \delta_i \leq \delta_i^{max} \tag{4.6}$$

4.3.2 AC Optimal power flow

An OPF-AC, like an OPF-DC, aims to minimize the operating costs of the EPS, hence (4.7) describes the objective function, which is subject to (4.8) and (4.9) that determine the balance of active and reactive power between generation, demand, and power flow, by (4.10) and (4.11) that determines the active and reactive power flow that circulates through a transmission line depending on its susceptance and conductance, in (4.12) and (4.13) the active and reactive power limit restrictions of the generators are shown, (4.14) delimits the voltages in each node and the limits of the angular difference in each branch is (4.15) (20)–(22).

$ObjetiveFunction:$

$$min\ Z = \sum_{g=1}^{N_G} C_g P_g \tag{4.7}$$

$Subjectto:$

$$\sum_{i=1}^{n_{bus}} \sum_{j=1}^{n_{bus}} P_{i-j} = P_{Gi} - P_{Di}; \;\; \forall \, i, j \;\in\; N_B \tag{4.8}$$

$$\sum_{i=1}^{n_{bus}} \sum_{j=1}^{n_{bus}} Q_{i-j} = Q_{Gi} - Q_{Di}; \;\; \forall \, i, j \;\in\; N_B \tag{4.9}$$

$$P_{i-j} = V_i^2 G_{i-j} - V_i V_j \left[G_{i-j} cos(\delta_{i-j}) - B_{i-j} sin(\delta_{i-j}) \right] \tag{4.10}$$

$$Q_{i-j} = V_i^2 B_{i-j} + V_i V_j \left[G_{i-j} sin(\delta_{i-j}) - B_{i-j} cos(\delta_{i-j}) \right] \tag{4.11}$$

$$P_{Gi}^{min} \leq P_{Gi} \leq P_{Gi}^{max} \tag{4.12}$$

$$Q_{Gi}^{min} \leq Q_{Gi} \leq Q_{Gi}^{max} \tag{4.13}$$

$$V_i^{min} \leq V_i \leq V_i^{max} \tag{4.14}$$

$$\delta_i^{min} \leq \delta_i \leq \delta_i^{max} \tag{4.15}$$

$$\delta_{i-j} = \delta_i - \delta_j \tag{4.16}$$

$$-SIL \leq S_{i-j} \leq SIL \tag{4.17}$$

4.4 Matlab - GAMS Coding and Results Analysis.

4.4.1 OPF-DC code

The following is the code developed in Matlab that allows the evaluation of the optimal DC power flows (OPF-DC), which has been divided into two parts. The first one is the main program in which the power system to be analyzed is configured, the sending and receiving of data between GAMS and Matlab, the file names to be generated, and the treatment of the OPF-DC results, which are Power flows in the lines, Node voltages, and Generation dispatch. The second part is the GAMS coding written in Matlab, which will be the OPF-DC code to be optimized.

Matlab main script code, written in a new script.

```
1  clc; clearvars; close all;
2  % Initialization of the interface with GAMS
3  GDXFileIN = 'Send'; %GDX File for send data to GAMS
4  GDXFileOUT = 'Get'; %GDX File for call data from GAMS
5  GMSFile = 'OPF_DC_30bus';
6  % The GAMS code for OPF-DC is written.
7  file='30BARRAS.xlsx'
8  writegams_OPFDC(GMSFile,file);
9  %% OPF-DC results are extracted
10 system(string(strcat('gams',{' '},GMSFile,' lo=3 gdx=',GDXFileOUT)))
       ;
11 %
       **************************************************************************

12 % Power flows in the lines
13 Pij.name = 'Pij';
14 Pij.compress = 'true';
15 Pij.form = 'sparse';
16 Pij = rgdx(GDXFileOUT,Pij);
17 %
       **************************************************************************

18 % Voltage angle at nodes
19 delta.name = 'delta';
20 delta.compress = 'true';
21 delta.form = 'sparse';
22 delta = rgdx(GDXFileOUT,delta);
23 %
       **************************************************************************

24 % Generation dispatch
25 Pgen.name = 'Pgen';
26 Pgen.compress = 'true';
27 Pgen.form = 'sparse';
28 Pgen = rgdx(GDXFileOUT,Pgen);
```

Line 7 of the main code represents the file in which the data is stored; for the case study, we have used as an example the data of the IEEE 30−busbar system in

an Excel file (30NODES.xlsx), which contains 3 sheets, called: data_gen, demand and lines, all sheets must start in cell A1. Electric demand is in the **demand** sheet and is shown in table 4.2, line data is in the **lines** sheet and is shown in table 4.3, and generator data is in the **data_gen** sheet, which is shown in table 4.4.

Table 4.2: IEEE 30busbar system electrical demand data

	Pd	**Qd**
1	0	0
2	21,7	12,7
3	2,4	1,2
4	7,6	1,6
5	94,2	19
6	0	0
7	22,8	10,9
8	30	30
9	0	0
10	5,8	2
11	5	2
12	11,2	7,5
13	0	0
14	6,2	1,6
15	8,2	2,5
16	3,5	1,8
17	9	5,8
18	3,2	0,9
19	9,5	3,4
20	2,2	0,7
21	17,5	11,2
22	0	0
23	3,2	1,6
24	8,7	6,7
25	0	0
26	3,5	2,3
27	0	0
28	0	0
29	2,4	0,9
30	10,6	1,9

Table 4.3: IEEE 30busbar system lines data

	r	x	b	sil
1 . 2	0,02	0,0575	0	130,00
1 . 3	0,05	0,1852	0	130,00
2 . 4	0,06	0,1737	0	65,00
3 . 4	0,01	0,0379	0	130,00
2 . 5	0,05	0,02	0	130,00
2 . 6	0,06	0,1763	0	65,00
4 . 6	0,01	0,0414	0	90,00
5 . 7	0,05	0,116	0	70,00
6 . 7	0,03	0,082	0	130,00
6 . 8	0,01	0,09	0	32,00
6 . 9	0	0,208	0	65,00
6 . 10	0	0,556	0	32,00
9 . 11	0	0,208	0	65,00
9 . 10	0	0,11	0	65,00
4 . 12	0	0,23	0	65,00
12 . 13	0	0,14	0	65,00
12 . 14	0,13	0,2559	0	32,00
12 . 15	0,0662	0,1304	0	32,00
12 . 16	0,0945	0,12	0	32,00
14 . 15	0,221	0,12	0	16,00
16 . 17	0,0824	0,1932	0	16,00
15 . 18	0,107	0,2185	0	16,00
18 . 19	0,0639	0,1292	0	16,00
19 . 20	0,034	0,068	0	32,00
10 . 20	0,0936	0,209	0	32,00
10 . 17	0,0324	0,0845	0	32,00
10 . 21	0,0348	0,0749	0	32,00
10 . 22	0,0727	0,1499	0	32,00
21 . 22	0,0116	0,22	0	32,00
15 . 23	0,1	0,202	0	16,00
22 . 24	0,115	0,18	0	16,00
23 . 24	0,132	0,27	0	16,00
24 . 25	0,1885	0,3292	0	16,00
25 . 26	0,2544	0,38	0	16,00
25 . 27	0,1093	0,2087	0	16,00
28 . 27	0	0,369	0	65,00
27 . 29	0,2198	0,4153	0	16,00
27 . 30	0,3202	0,6027	0	16,00
29 . 30	0,2399	0,453	0	16,00
8 . 28	0,0636	0,2	0	32,00
6 . 28	0,0169	0,06	0	32,00

Table 4.4: IEEE 30busbar system generators data

	pmax	pmin	b	Qmax	Qmin
1	120	0	2	150	0
2	120	0	1,75	60	-20
5	100	0	1	44,7	-15
8	110	0	3,25	62,5	-15
11	60	0	3	40	-10
13	80	0	3	48,7	-15

Function writing GAMS code developed in Matlab, written in a new function called writegams_OPFDC.m.

```
function writegams_OPFDC(FileGMS,file)
if ~contains(FileGMS,'.gms')
    FileGMS = strcat(FileGMS,'.gms');
end
filesDir = dir;
s0 = 0;
for c0=1:length(filesDir)
    findGMS = strfind(string(filesDir(c0).name),FileGMS);
    if isempty(findGMS)&&s0~=1 %#ok<*STREMP>
        s0 = 0;
    else
        s0 = 1;
    end
end
% s0 = 0: File does not exist .gms
% s0 = 1: File exists .gms
%*********************************************************************
if s0 == 0
    fgms = fopen(FileGMS,'w');
    clc;
    fprintf(fgms,'$Title Optimal power flow DC - 30 Bus System \n\n'
       );
    fprintf(fgms,'$offsymxref \n');
    fprintf(fgms,'$offsymlist \n\n');
%*********************************************************************
    fprintf(fgms,'option limrow=0; \n');
    fprintf(fgms,'option limcol=0; \n');
    fprintf(fgms,'option solprint=on; \n');
    fprintf(fgms,'option sysout=off; \n\n');
%*********************************************************************
    fprintf(fgms,'option LP=CPLEX; \n');
    fprintf(fgms,'option MIP=CPLEX; \n');
    fprintf(fgms,'option NLP=CONOPT; \n');
    fprintf(fgms,'option MINLP=COINBONMIN; \n\n');
%*********************************************************************
    fprintf(fgms,'option OPTCR=0; \n');
    fprintf(fgms,'option OPTCA=0; \n\n');
%*********************************************************************
```

```
38      fprintf(fgms,'SCALAR \n');
39      fprintf(fgms,'sbase     /100/ \n');
40      fprintf(fgms,'pi        /3.141592654/; \n\n');
41  %*********************************************************************
42      fprintf(fgms,'set \n');
43      fprintf(fgms,'bus               /1*30/ \n');
44      fprintf(fgms,'slack(bus)        /1/; \n\n');
45      fprintf(fgms,'alias(bus,node); \n');
46  % i=M:\\OPFDC\\%s corresponds to the address of the file where the
47  % EPS data is stored
48  %*********************************************************************
49      fprintf(fgms,'TABLE    data_gen(bus,*)  generators data \n');
50      fprintf(fgms,'$call =xls2gms r=data_gen!A1:F7 i=M:\\OPFDC\\%s O=
            Buses.inc \n',file);
51      fprintf(fgms,'$include Buses.inc \n');
52      fprintf(fgms,'; \n\n');
53  %*********************************************************************
54      fprintf(fgms,'TABLE    demand(bus,*)  electrical demand data \n'
            );
55      fprintf(fgms,'$call =xls2gms r=demand!A1:C31 i=M:\\OPFDC\\%s O=
            Buses.inc \n',file);
56      fprintf(fgms,'$include Buses.inc \n');
57      fprintf(fgms,'; \n\n');
58  %*********************************************************************
59      fprintf(fgms,'TABLE    lines(bus,node,*)  lines data \n');
60      fprintf(fgms,'$call =xls2gms r=lines!A1:E42 i=M:\\OPFDC\\%s O=
            Buses.inc \n',file);
61      fprintf(fgms,'$include Buses.inc \n');
62      fprintf(fgms,'; \n\n');
63  %*********************************************************************
64      fprintf(fgms,'display lines, demand, data_gen; \n');
65  %*********************************************************************
66      fprintf(fgms,'lines(bus,node,''x'')$(lines(bus,node,''x'')=0) =
            lines(node,bus,''x''); \n');
67      fprintf(fgms,'lines(bus,node,''sil'')$(lines(bus,node,''sil'')
            =0) = lines(node,bus,''sil''); \n');
68      fprintf(fgms,'lines(bus,node,''bij'')$lines(bus,node,''sil'') =
            1/lines(bus,node,''x''); \n');
69  %*********************************************************************
70      fprintf(fgms,'Parameter connection_lin(bus,node); \n');
71      fprintf(fgms,'connection_lin(bus,node)$(lines(bus,node,''sil'')
            and lines(node,bus,''sil'')) = 1; \n');
72      fprintf(fgms,'connection_lin(bus,node)$(connection_lin(node,bus)
            ) = 1; \n');
73      fprintf(fgms,'display connection_lin, lines, demand, data_gen; \
            n\n');
74  %*********************************************************************
75      fprintf(fgms,'Variable Z, Pij(bus,node), Pgen(bus), delta(bus) ;
             \n');
76      fprintf(fgms,'Equation res1, res2, res3; \n\n');
77  %*********************************************************************
78      fprintf(fgms,'res1.. \n');
```

```
79      fprintf(fgms,'Z =e= sum((bus), Pgen(bus)*data_gen(bus,''b'')*
            sbase); \n\n');
80  %*********************************************************************
81      fprintf(fgms,'res2(bus,node)$connection_lin(bus,node).. \n');
82      fprintf(fgms,'Pij(bus,node) =e= lines(bus,node,''bij'')*(delta(
            bus)-delta(node)); \n\n');
83  %*********************************************************************
84      fprintf(fgms,'res3(bus).. \n');
85      fprintf(fgms,'Pgen(bus)$data_gen(bus,''Pmax'') - demand(bus,''pd
            '')/base =e= sum(node$connection_lin(node,bus), Pij(bus,node)
            ); \n\n');
86  %*********************************************************************
87      fprintf(fgms,'Model loadflow /all/; \n\n');
88  %*********************************************************************
89      fprintf(fgms,'Pgen.lo(bus) = data_gen(bus,''Pmin'')/sbase; \n');
90      fprintf(fgms,'Pgen.up(bus) = data_gen(bus,''Pmax'')/sbase; \n');
91  %*********************************************************************
92      fprintf(fgms,'delta.up(bus) = pi/2; \n');
93      fprintf(fgms,'delta.lo(bus) =-pi/2; \n');
94      fprintf(fgms,'delta.l(bus) = 0; \n');
95      fprintf(fgms,'delta.fx(slack) = 0; \n\n');
96  %*********************************************************************
97      fprintf(fgms,'Pij.up(bus,node)$((connection_lin(bus,node))) = 1*
            lines(bus,node,''sil'')/base; \n');
98      fprintf(fgms,'Pij.lo(bus,node)$((connection_lin(bus,node))) =-1*
            lines(bus,node,''sil'')/base; \n');
99      fprintf(fgms,'solve load flow min Z using lp; \n');
100 end
```

It is important to remember that the main script, the function, and the Excel file must be in the same location.

4.4.2 OPF-AC code

The following is the code developed in Matlab that allows the evaluation of the optimal AC power flows (OPF-AC), which has been divided into 2 parts. The first one is the main program in which the power system to be analyzed is configured, the sending and receiving of data between GAMS and Matlab, the file names to be generated, and the treatment of the OPF-AC results, which are Power flows in the lines, Node voltages, and Generation dispatch. The second part is the GAMS coding written in Matlab, which will be the OPF-AC code to be optimized.

Matlab main script code, written in a new script.

```
1  clc; clearvars; close all;
2  % Initialization of the interface with GAMS
3  GDXFileIN = 'Send'; %GDX File for send data to GAMS
4  GDXFileOUT = 'Get'; %GDX File for call data from GAMS
5  GMSFile = 'OPF_AC_30bus';
6  % The GAMS code for OPF-AC is written.
7  file='30NODES.xlsx'
```

```
writegams_OPF(GMSFil,file);
%% OPF-AC results are extracted
system(string(strcat('gams',{' '},GMSFile,' lo=3 gdx=',GDXFileOUT)))
    ;
%*********************************************************************
%  Power flows in the lines
Pij.name = 'Pij';
Pij.compress = 'true';
Pij.form = 'sparse';
Pij = rgdx(GDXFileOUT,Pij);
Qij.name = 'Qij';
Qij.compress = 'true';
Qij.form = 'sparse';
Qij = rgdx(GDXFileOUT,Qij);
%*********************************************************************
% Node voltages
V.name = 'V';
V.compress = 'true';
V.form = 'sparse';
V = rgdx(GDXFileOUT,V);
delta.name = 'delta';
delta.compress = 'true';
delta.form = 'sparse';
delta = rgdx(GDXFileOUT,delta);
%*********************************************************************
% Generation dispatch
Pgen.name = 'Pgen';
Pgen.compress = 'true';
Pgen.form = 'sparse';
Pgen = rgdx(GDXFileOUT,Pgen);
Qgen.name = 'Qgen';
Qgen.compress = 'true';
Qgen.form = 'sparse';
Qgen = rgdx(GDXFileOUT,Qgen);
```

Function writing GAMS code developed in Matlab, written in a new function called writegams_OPF.m, and the EPS data is the same file used in OPF-DC (30NODES.xlsx).

```
function writegams_OPF(FileGMS,file)
if ~contains(FileGMS,'.gms')
    FileGMS = strcat(FileGMS,'.gms');
end
filesDir = dir;
s0 = 0;
for c0=1:length(filesDir)
    findGMS = strfind(string(filesDir(c0).name),FileGMS);
    if isempty(findGMS)&&s0~=1 %#ok<*STREMP>
        s0 = 0;
    else
        s0 = 1;
    end
```

```
14 end
15 % s0 = 0: File does not exist .gms
16 % s0 = 1: File exists .gms
17 %*********************************************************************
18 if s0 == 0
19     fgms = fopen(FileGMS,'w');
20     clc;
21     fprintf(fgms,'$Title Optimal power flow AC - 30 Bus System \n\n'
           );
22     fprintf(fgms,'$offsymxref \n');
23     fprintf(fgms,'$offsymlist \n\n');
24     fprintf(fgms,'option limrow=0; \n');
25     fprintf(fgms,'option limcol=0; \n');
26     fprintf(fgms,'option solprint=on; \n');
27     fprintf(fgms,'option sysout=off; \n\n');
28     fprintf(fgms,'option LP=CPLEX; \n');
29     fprintf(fgms,'option MIP=CPLEX; \n');
30     fprintf(fgms,'option NLP=CONOPT; \n');
31     fprintf(fgms,'option MINLP=COINBONMIN; \n\n');
32     fprintf(fgms,'option OPTCR=0; \n');
33     fprintf(fgms,'option OPTCA=0; \n\n');
34 %*********************************************************************
35     fprintf(fgms,'SCALAR \n');
36     fprintf(fgms,'sbase    /100/ \n');
37     fprintf(fgms,'pi       /3.141592654/; \n\n');
38 %*********************************************************************
39     fprintf(fgms,'set \n');
40     fprintf(fgms,'bus              /1*30/ \n');
41     fprintf(fgms,'slack(bus)       /1/; \n\n');
42     fprintf(fgms,'alias(bus,node); \n');
43 % i=M:\\OPFAC\\%s corresponds to the address of the file where the
44 % EPS data is stored
45 %*********************************************************************
46     fprintf(fgms,'TABLE    data_gen(bus,*)  generators data \n');
47     fprintf(fgms,'$call =xls2gms r=data_gen!A1:F7 i=M:\\OPFAC\\%s O=
           Buses.inc \n',file);
48     fprintf(fgms,'$include Buses.inc \n');
49     fprintf(fgms,'; \n\n');
50 %*********************************************************************
51     fprintf(fgms,'TABLE    demand(bus,*)  electrical demand data \n'
           );
52     fprintf(fgms,'$call =xls2gms r=demand!A1:C31 i=M:\\OPFAC\\%s O=
           Buses.inc \n',file);
53     fprintf(fgms,'$include Buses.inc \n');
54     fprintf(fgms,'; \n\n');
55 %*********************************************************************
56     fprintf(fgms,'TABLE    lines(bus,node,*)  lines data \n');
57     fprintf(fgms,'$call =xls2gms r=lines!A1:E42 i=M:\\OPFAC\\%s O=
           Buses.inc \n',file);
58     fprintf(fgms,'$include Buses.inc \n');
59     fprintf(fgms,'; \n\n');
60 %*********************************************************************
```

```
61      fprintf(fgms,'display lines, demand, data_gen; \n');
62  %*********************************************************************
63      fprintf(fgms,'lines(bus,node,''x'')$(lines(bus,node,''x'')=0) =
           lines(node,bus,''x''); \n');
64      fprintf(fgms,'lines(bus,node,''r'')$(lines(bus,node,''r'')=0) =
           lines(node,bus,''r''); \n');
65      fprintf(fgms,'lines(bus,node,''b'')$(lines(bus,node,''b'')=0) =
           lines(node,bus,''b''); \n');
66      fprintf(fgms,'lines(bus,node,''sil'')$(lines(bus,node,''sil'')
           =0) = lines(node,bus,''sil''); \n');
67      fprintf(fgms,'lines(bus,node,''bij'')$lines(bus,node,''sil'') =
           1/lines(bus,node,''x''); \n');
68      fprintf(fgms,'lines(bus,node,''z'')$lines(bus,node,''sil'') =
           sqrt(sqr(lines(bus,node,''x'')) + sqr(lines(bus,node,''r'')))
           ; \n');
69      fprintf(fgms,'lines(node,bus,''z'')$(lines(bus,node,''z'')=0) =
           lines(bus,node,''z''); \n');
70      fprintf(fgms,'lines(bus,node,''th'')$(lines(bus,node,''sil'')
           and lines(bus,node,''x'') and lines(bus,node,''r'')) = arctan
           (lines(bus,node,''x'')/(lines(bus,node,''r''))); \n');
71      fprintf(fgms,'lines(bus,node,''ybus'')$lines(bus,node,''sil'') =
            1/lines(bus,node,''z''); \n');
72      fprintf(fgms,'lines(node,bus,''ybus'')$(lines(bus,node,''ybus'')
           =0) = lines(bus,node,''ybus''); \n\n');
73  %*********************************************************************
74      fprintf(fgms,'Parameter connection_lin(bus,node); \n');
75      fprintf(fgms,'connection_lin(bus,node)$(lines(bus,node,''sil'')
           and lines(node,bus,''sil'')) = 1; \n');
76      fprintf(fgms,'connection_lin(bus,node)$(connection_lin(node,bus)
           ) = 1; \n');
77      fprintf(fgms,'display connection_lin, lines, demand, data_gen; \
           n\n');
78  %*********************************************************************
79      fprintf(fgms,'Variable Z, Pij(bus,node), Qij(bus,node), Pgen(bus
           ), Qgen(bus), delta(bus), V(bus); \n');
80      fprintf(fgms,'Equation res1, res2, res3, res4, res5; \n\n');
81  %*********************************************************************
82      fprintf(fgms,'res1.. \n');
83      fprintf(fgms,'Z =g= sum((bus), Pgen(bus)*data_gen(bus,''b'')*
           sbase); \n\n');
84  %*********************************************************************
85      fprintf(fgms,'res2(bus,node)$connection_lin(bus,node).. \n');
86      fprintf(fgms,'Pij(bus,node) =e= (V(bus)*V(bus)*cos(lines(node,
           bus,''th'')) - V(bus)*V(node)*cos(delta(bus)-delta(node)+
           lines(node,bus,''th'')))*lines(node,bus,''ybus''); \n\n');
87  %*********************************************************************
88      fprintf(fgms,'res3(bus,node)$connection_lin(bus,node).. \n');
89      fprintf(fgms,'Qij(bus,node) =e= (V(bus)*V(bus)*sin(lines(node,
           bus,''th'')) - V(bus)*V(node)*sin(delta(bus) - delta(node) +
           lines(node,bus,''th'')))*lines(node,bus,''ybus'') - lines(
           node,bus,''b'')*V(bus)*V(bus)/2; \n\n');
90  %*********************************************************************
```

```
91      fprintf(fgms,'res4(bus).. \n');
92      fprintf(fgms,'Pgen(bus)$data_gen(bus,''Pmax'') - demand(bus,''pd
            '')/base =e= sum(node$connection_lin(node,bus), Pij(bus,node)
            ); \n\n');
93  %*********************************************************************
94      fprintf(fgms,'res5(bus).. \n');
95      fprintf(fgms,'Qgen(bus)$data_gen(bus,''Qmax'') - demand(bus,''qd
            '')/base =e= sum(node$connection_lin(node,bus), Qij(bus,node)
            ); \n\n');
96  %*********************************************************************
97      fprintf(fgms,'Model loadflow /all/; \n\n');
98  %*********************************************************************
99      fprintf(fgms,'Pgen.lo(bus) = data_gen(bus,''Pmin'')/sbase; \n');
100     fprintf(fgms,'Pgen.up(bus) = data_gen(bus,''Pmax'')/sbase; \n');
101     fprintf(fgms,'Qgen.lo(bus) = data_gen(bus,''Qmin'')/sbase; \n');
102     fprintf(fgms,'Qgen.up(bus) = data_gen(bus,''Qmax'')/sbase; \n\n'
            );
103 %*********************************************************************
104     fprintf(fgms,'delta.up(bus) = pi/2; \n');
105     fprintf(fgms,'delta.lo(bus) =-pi/2; \n');
106     fprintf(fgms,'delta.l(bus) = 0; \n');
107     fprintf(fgms,'delta.fx(slack) = 0; \n\n');
108 %*********************************************************************
109     fprintf(fgms,'Pij.up(bus,node)$((connection_lin(bus,node))) = 1*
            lines(bus,node,''sil'')/sbase; \n');
110     fprintf(fgms,'Pij.lo(bus,node)$((connection_lin(bus,node))) =-1*
            lines(bus,node,''sil'')/sbase; \n');
111     fprintf(fgms,'Qij.up(bus,node)$((connection_lin(bus,node))) = 1*
            lines(bus,node,''sil'')/sbase; \n');
112     fprintf(fgms,'Qij.lo(bus,node)$((connection_lin(bus,node))) =-1*
            lines(bus,node,''sil'')/sbase; \n\n');
113 %*********************************************************************
114     fprintf(fgms,'V.lo(bus) = 0.9; \n');
115     fprintf(fgms,'V.up(bus) = 1.1; \n');
116     fprintf(fgms,'V.l(bus) = 1; \n');
117     fprintf(fgms,'V.fx(slack) = 1; \n\n');
118     fprintf(fgms,'solve load flow min Z using nlp; \n');
119 end
```

In each of the codes in the main script, the treatment of the variables resulting from the optimization process, such as voltage, angle, and active and reactive power, is carried out at the end.

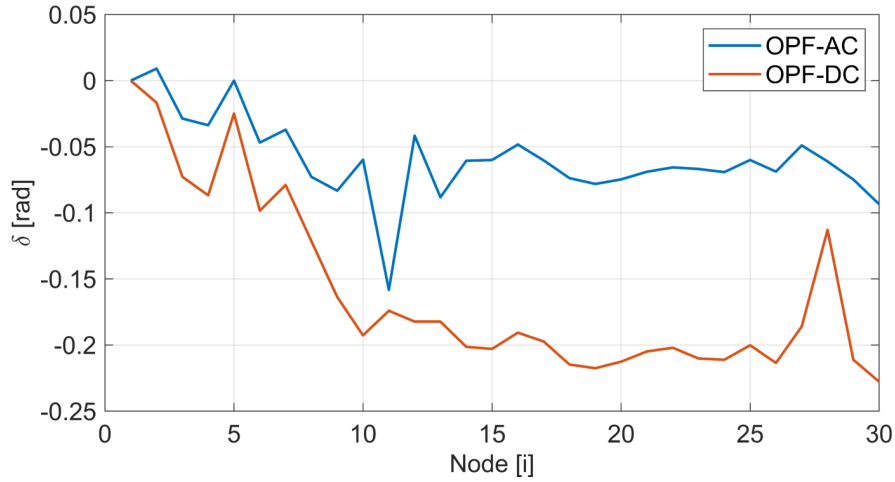


Figure 4.9: δ OPF$-$AC vs δ OPF$-$DC

Figure 4.9 shows the comparison of the results of the voltage angle optimization in each of the EPS nodes; as can be seen, there is much similarity in the results in terms of the trend, but because the OPF-DC gives approximate results, its fluctuations are more significant because there are fewer electrical parameters to analyze or that restrict the optimization model.

This process can be carried out with each variable, and even more complex analyses can be started, such as studies of different stability indexes.

4.5 Conclusions

The use of computational tools for the study and analysis of power systems has led to the availability of new methodologies that can be more robust and that find solutions to problems that could not be solved before. For example, the use of OPF in large systems some years ago was solved by continuous power Flow, which is a modification of the classical Newton - Raphso model, and now it can be solved by linear or nonlinear optimization.

Having an interface between GAMS and Matlab facilitates the EPS analysis processes since one can generate different conditions for the model in Matlab and will always call the same optimization subroutine. This situation is very complex to perform directly in GAMS.

Having generic methodologies to solve optimal EPS power flows allows finding solutions that simulators do not yet have in their algorithm; for example, optimal switching of transmission lines is a new theory that is already applied to date in EPS in contingency scenarios and avoids possible blackouts by not having a

characterization of the operation in this type of scenarios and now having an answer based on optimization criteria the more reliable solutions.

The interface in Matlab and GAMS reduces the analysis and study times of the EPS, simplifies the traditional way of performing such research, and the reliability of not having errors in the manipulation of the information by not having external files, as a result of independent analysis, is increased.

Bibliography

[1] D. Carrión, J. W. González, G. J. López, and I. A. Isaac, "Alternative fault detection method in electrical power systems based on ARMA model", *2019 FISE-IEEE/CIGRE Conference - Living the Energy Transition, FISE/CIGRE 2019*, 2019. DOI: 10.1109/FISECIGRE48012.2019.8984981.

[2] J. Toctaquiza and D. Carrión, "Estado del arte modelo óptimo de operación posterior a ataques intencionales considerando conmutación de los sistemas de transmisión", *ITECKNE Innovación e Investigación en Ingeniería*, vol. 18, no. 2, p. 17, 2021, ISSN: 1692-1798. DOI: https://doi.org/10.15332/iteckne.v18i2.2559.

[3] D. Carrión, J. Francisco, and M. Paul, "Revisión para la restauración optima de la operación del sistema eléctrico basado en criterios de calidad de energía y estabilidad", *Revista de I+D Tecnológico*, vol. 17, no. 1, p. 9, 2021. DOI: 10.33412/idt.v17.1.2928.

[4] J. Pilatásig Lasluisa and D. Carrión, "Resiliencia de Sistemas Eléctricos de Potencia mediante la Conmutación de Líneas de Transmisión – Estado del arte", *I+D Tecnológico*, vol. 16, no. 2, 2020, ISSN: 1680-8894. DOI: 10.33412/idt.v16.2.2834.

[5] J. Ramirez, D. Carrión, and E. Inga, "Compensación reactiva en redes eléctricas de transmisión basado en programación no lineal considerando ubicación óptima de SVC", *Revista de I+D Tecnológico*, vol. 17, no. February, 2021. DOI: 10.33412/idt.v17.1.2918.

[6] J. Palacios and D. F. Carrión Galarza, "Estado del arte de la planeación de expansión de sistemas de transmisión", *I+D Tecnológico*, vol. 16, no. 2, pp. 1–8, 2020. DOI: 10.33412/idt.v16.2.2835.

[7] N. Gupta, R. Shekhar, and P. K. Kalra, "Computationally efficient composite transmission expansion planning: A Pareto optimal approach for techno-economic solution", *International Journal of Electrical Power & Energy Systems*, vol. 63, pp. 917–926, Dec. 2014, ISSN: 01420615. DOI: 10.1016/j.ijepes.2014.05.070. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S014206151400341X.

[8] A. R. Kumar and L. Premalatha, "Electrical Power and Energy Systems Optimal power flow for a deregulated power system using adaptive real coded biogeography-based optimization", *INTERNATIONAL JOURNAL OF ELECTRICAL POWER AND ENERGY SYSTEMS*, vol. 73, pp. 393–399, 2015, ISSN: 0142-0615. DOI: 10.1016/j.ijepes.2015.05.011.

[9] R. S. Ferreira, C. L. T. Borges, and M. V. F. Pereira, "A flexible mixed-integer linear programming approach to the AC optimal power flow in distribution systems", *IEEE Transactions on Power Systems*, vol. 29, no. 5, pp. 2447–2459, 2014, ISSN: 08858950. DOI: 10.1109/TPWRS.2014.2304539.

[10] T. Akbari and M. Tavakoli Bina, "Linear approximated formulation of AC optimal power flow using binary discretisation", *IET Generation, Transmission & Distribution*, vol. 10, no. 5, pp. 1117–1123, 2016, ISSN: 1751-8687. DOI: 10.1049/iet-gtd.2015.0388. [Online]. Available: http://digital-library.theiet.org/content/journals/10.1049/iet-gtd.2015.0388.

[11] V. Sarkar and S. a. Khaparde, "Implementation of LMP-FTR mechanism in an AC-DC system", *IEEE Transactions on Power Systems*, vol. 23, no. 2, pp. 737–746, 2008, ISSN: 08858950. DOI: 10.1109/TPWRS.2008.920202.

[12] A. Kargarian, J. Mohammadi, J. Guo, *et al.*, "Toward Distributed/Decentralized DC Optimal Power Flow Implementation in Future Electric Power Systems", *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 2574–2594, 2018, ISSN: 19493053. DOI: 10.1109/TSG.2016.2614904.

[13] "Pricing Energy and Ancillary Services in Integrated Market Systems by an Optimal Power Flow", *IEEE Transactions on Power Systems*, vol. 19, no. 1, pp. 339–347, 2004, ISSN: 0885-8950. DOI: 10.1109/TPWRS.2003.820701.

[14] B. Stott, J. Jardim, and O. Alsac, "DC Power Flow Revisited", *IEEE Transactions on Power Systems*, vol. 24, no. 3, pp. 1290–1300, 2009, ISSN: 0885-8950. DOI: 10.1109/TPWRS.2009.2021235.

[15] M. Bachtiar Nappu, A. Arief, and R. C. Bansal, "Transmission management for congested power system: A review of concepts, technical challenges and development of a new methodology", *Renewable and Sustainable Energy Reviews*, vol. 38, pp. 572–580, Oct. 2014, ISSN: 13640321. DOI: 10.1016/j.rser.2014.05.089. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1364032114004237.

[16] T. Akbari and M. Tavakoli Bina, "A linearized formulation of AC multi-year transmission expansion planning: A mixed-integer linear programming approach", *Electric Power Systems Research*, vol. 114, pp. 93–100, Sep. 2014, ISSN: 03787796. DOI: 10.1016/j.epsr.2014.04.013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378779614001539.

[17] P. Masache, D. Carrión, and J. Cárdenas, "Optimal Transmission Line Switching to Improve the Reliability of the Power System Considering AC Power Flows", *Energies 2021, Vol. 14, Page 3281*, vol. 14, no. 11, p. 3281, Jun. 2021. DOI: 10.3390/EN14113281. [Online]. Available: https://www.mdpi.com/1137108.

[18] S. Pinzón, D. Carrión, and E. Inga, "Optimal Transmission Switching Considering N-1 Contingencies on Power Transmission Lines", *IEEE Latin America Transactions*, vol. 19, no. 4, pp. 534–541, 2021. DOI: 10.1109/TLA.2021.9448535. [Online]. Available: https://latamt.ieeer9.org/index.php/transactions/article/view/3691/636.

[19] D. Carrión, J. Palacios, M. Espinel, and J. W. González, "Transmission Expansion Planning Considering Grid Topology Changes and N-1 Contingencies Criteria", in *Recent Advances in Electrical Engineering, Electronics and Energy*, Springer, Ed., Springer, 2021, pp. 266–279. DOI: 10.1007/978-3-030-72208-1_20. [Online]. Available: http://link.springer.com/10.1007/978-3-030-72208-1%7B%5C_%7D20.

[20] D. Carrión, E. García, M. Jaramillo, and J. W. González, "A Novel Methodology for Optimal SVC Location Considering N-1 Contingencies and Reactive Power Flows Reconfiguration", *Energies*, vol. 14, no. 20, pp. 1–17, 2021. DOI: 10.3390/en14206652.

[21] A. Lemus, D. Carrión, E. Aguire, and J. W. Gonz, "Location of distributed resources in rural-urban marginal power grids considering the voltage collapse prediction index", *Ingenius*, vol. 28, pp. 25–33, 2022. DOI: https://doi.org/10.17163/ings.n28.2022.02.

[22] F. Quinteros, D. Carrión, and M. Jaramillo, "Optimal Power Systems Restoration Based on Energy Quality and Stability Criteria", *Energies*, vol. 15, no. 6, 2022. DOI: 10.3390/en15062062.

5. Innovative Control Paradigms for DC Motors Control

Ph.D. Wilson Pavón Vallejos is a Professor of Mechatronic, Architecture Bachelor, and Automation and Control Postgraduate in the Universidad Politécnica Salesiana (Ecuador)
Email: wpavon@ups.edu.ec
https://orcid.org/0000-0002-9319-8815
DOI: 10.17163/abyaups.44.350

5.1 Introduction

This research proposes innovative control paradigms for DC motors control, integrating variable coefficient fractional-order proportional-integral-derivative (PID), model predictive control (MPC), and State Feedback with Ackerman and Bessel Polynomial Strategies for Optimal Performance. The fractional order is utilized to enhance the control performance of the system. The optimal tuning of the proposed controller is achieved through a hybrid optimization algorithm that combines the particle swarm optimization (PSO) and the genetic algorithm (GA). The PSO algorithm searches for the optimal set of proportional, integral, and derivative gains, while the GA is employed to optimize the fractional order and the controller coefficients. The hybrid algorithm can efficiently obtain the optimal controller parameters and fractional order with a small number of iterations. The proposed variable coefficient fractional order PID controller can be applied to several benchmark control problems, including the inverted pendulum, magnetic levitation system, and water level control. The simulation results show that the proposed controller outperforms the conventional PID controller and the fixed coefficient fractional order PID controller regarding tracking accuracy, disturbance rejection, and robustness. The

proposed controller also exhibits good robustness against parameter variations and external disturbances. This research paper proposes a novel variable coefficient fractional order PID controller design with optimal tuning. The article studies the structure of variable-order fractional proportional-integral-derivative (VFPID) controllers for linear dynamical systems. The proposed VFPID controller is designed to regulate the joint variables of a mobile manipulator. The significant contribution of the paper is to fill this gap by presenting a novel approach. To justify the claimed efficiency of the proposed VFPID controller, an improved VFPID controller is also offered in this paper, building the nonlinear relation between the integral gain Ki and derivative gain Kd. The proposed neural network (NNFPID) controller's parameters consist of derivative, integral, and proportional gains in addition to fractional integral and fractional derivative orders. The chapter is organized as follows, section 5.2 describes the related work, section 5.3 has the methodology, section 5.4 is the results and analysis, and Section 5.5 has the conclusions.

5.2 Related Works

Figure 5.1 illustrates the key authors who have published notable works on fractional PID applied to electrical problems, as observed during the literature review revision.

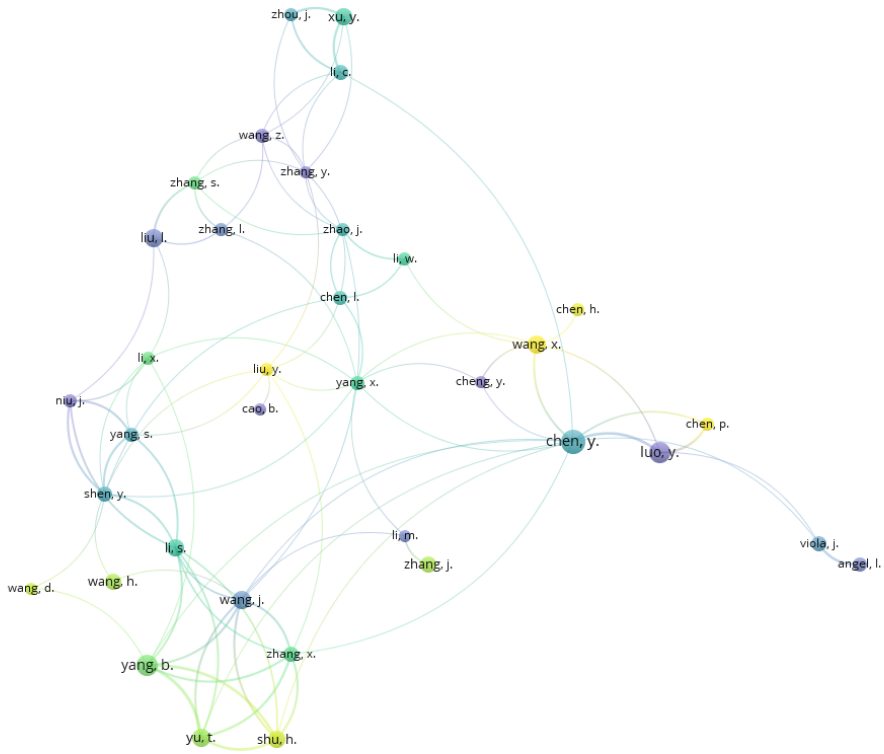


Figure 5.1: Bibliometric revision. Main authors that have published for fractional PID

The article (1) aims to address the advantages of fractional-order PID (FOPID) controllers over conventional integer-order (IO) PID controllers in terms of complexity and implementation. It reviews recent literature, provides analysis, and offers recommendations to understand the benefits of using FOPID controllers despite approximation challenges. Podlubny demonstrated that fractional order PID performs better than integer order PID controllers for fractional order dynamic control systems (2). Therefore, this research aims to improve the performance of mobile manipulators by proposing a new formulation of fractional order proportional integral (PI)/proportional integral derivative (PID) controller (3), (4). This research presents an optimal fuzzy proportional-integral-derivative control for a two-wheeled self-balancing robot (5). The paper of (6) deals with using the graphical tool of the diagrams of Bode, which is used to visualize the required actions so that the proposed systems are stable by including various controllers like PI or PID. The research (7)proposes an optimized Maximum Power Point Tracking (MPPT) controller based on the Grey Wolf Optimization algorithm for Solar photovoltaic systems. The proposed controller is compared to traditional Perturb, Observe, Incremental Conductance, and four metaheuristic algorithms. The models are analyzed for constant and variable irradiations, temperatures, and loads. The comparative results show that the MPPT controller optimized by the Grey Wolf Optimizer algorithm has superior performance, giving an average 6% output power higher than the other controllers under the test scenarios evaluated. The efficiency of the proposed model was, on average, 3% higher than the Incremental Conductance and Perturb & Observe controllers.

The paper (8) reviews scenario analysis methods to address uncertainty in modern power systems due to renewable energy sources and load variations. It covers scenario generation methods, scenario reduction methods, and scenario quality evaluation indices. The paper discusses new trends in scenario analysis methods and suggests potential research directions for scenario analysis of 100% renewable integrated power systems and integrated multiple energy systems.

The article (9) presents an experimental validation of fractional-order PID (FOPID) controllers on a two coupled tanks system. The paper (10) proposes a charging guiding strategy for electric vehicles (EVs) based on third-generation prospect theory (PT3) and a real-time coupling relationship model. The proposed method analyzes uncoordinated issues and balances the coupling system by guiding EVs based on a dynamic reference point and a full-state prospect description model. An algorithm based on a multi-time scale is used for real-time traffic distribution network simulation. The proposed method can coordinate the distribution of the charging load and relieve traffic pressure. The paper (11) presents a state feedback controller design for open-loop unstable systems using the sand cat swarm opti-

mization (SCSO) algorithm. The SCSO algorithm is a newly proposed metaheuristic algorithm that can efficiently find optimal solutions for optimization problems. The proposed SCSO-based state feedback controller optimizes control parameters with efficient convergence curve speed for three nonlinear control systems: an Inverted pendulum, a Furuta pendulum, and an Acrobat robot arm. Simulation results show that the proposed control method outperforms or has competitive effects compared to well-known metaheuristic-based algorithms. This research (12) proposes an adaptive cooperative co-evolutionary differential evolution method, QGDECC, to schedule railway train delays effectively. The QGDECC utilizes the quantum variable decomposition strategy and increment mutation method to improve convergence speed and strengthen the algorithm's robustness. The proposed method is applied to railway train delay scheduling to eliminate the impact of train delays on the network and minimize the gap between the rescheduled and original train schedules. The experimental results on benchmark functions and actual train operation data show that QGDECC has higher adaptability, faster convergence speed, and accuracy. The paper (13) proposes using the Twin Delay Deep Deterministic (TD3) policy gradient method to optimize the classical Proportional Integral (PI) controller for DC motor speed control using Reinforcement Learning (RL). The TD3 algorithm learns the optimal PI controller dynamics from a simulation environment, incorporating a reward mechanism that minimizes the optimal control objective function. The performance of the RL-based PI controller is compared with integer and fractional order PI controllers tuned by popular metaheuristic optimization algorithms. The results show that the proposed RL-PI method improves control performance, demonstrating the benefits of RL in industrial control applications.

5.3 Methodology

The following practical problem is considered about the speed control of a DC motor. The equation expresses the transfer function relating the output speed to the input voltage.

$$\frac{\dot{\theta}}{V} = \frac{K}{(Js+b)(Ls+R)+K^2}$$

Where:

- J Motor inertia
- b Damping frequency of the mechanical system
- K Constant of electromotive force
- R Electrical resistance
- L Electrical inductance

The parameters below are well known; $J = 0.5, b = 0.01$ y $K = 0.5$.

A DC motor is subjected to a unit step input voltage (1V). Figure 5.2 shows the speed response to the step set.

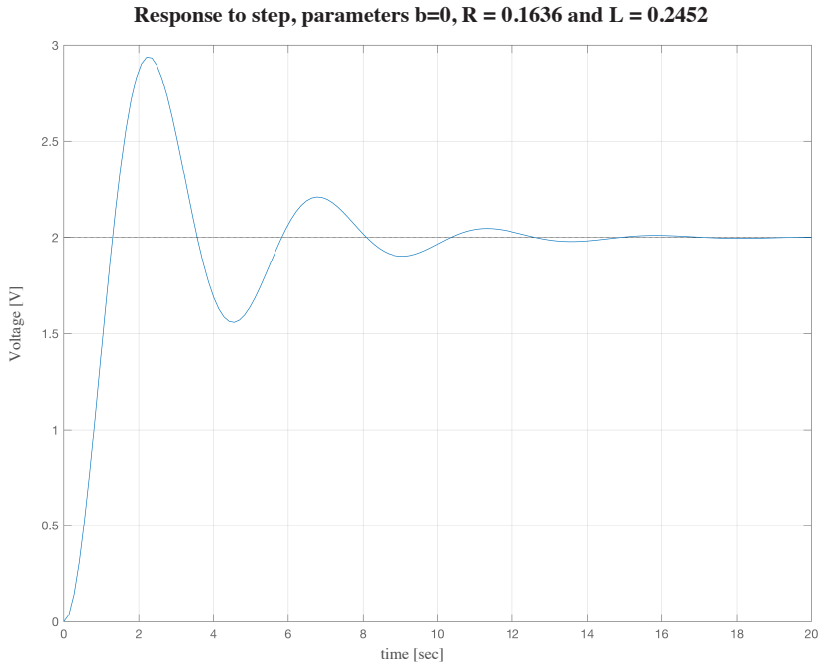


Figure 5.2: DC motor speed response to 1 V step input

Given these parameters, identify the model of the system and, thus, calculate the inductance and resistance of the DC motor. Tip: you have to fix equation 1 to a similar form of the general second-order equation. What is the possibility of simplifying the model assuming b=0? Try to compare it with the initial model. What impact does it have on calculating the value of R and L?

Based on the model obtained, the student must implement a PID controller as a second part of the problem. The controller parameters must be tuned by any method. It is required that the system response reaches a steady state as soon as possible without an overshoot greater than 10 %.

As the third part of the problem, based on the model obtained, the student should propose a state feedback controller to control the motor speed. This controller can be tuned with any method available in the literature. It is required that the system response reaches the steady state as soon as possible without an overshoot more significant than 10%.

Finally, the student must propose a control technique different from those previously offered. Similarly, the same design conditions should be imposed.

Submit a report and present the simulation of the identification and control in

SIMULINK/MATLAB.

5.3.1 First part: System Identification

The equation is:

$$G_1(s) = \frac{K_1}{(Js+b)(Ls+R)+K_1^2} \tag{5.1}$$

and the equation of a second-order system:

$$G_2(s) = \frac{K_2\omega^2}{s^2+2\zeta\omega s+\omega^2} \tag{5.2}$$

Developing equation (5.1) we have:

$$G_1(s) = \frac{K_1}{JLs^2+(JR+bL)s+(bR+K_1^2)} \tag{5.3}$$

$$G_1(s) = \frac{\frac{K_1}{JL}}{s^2+\frac{(JR+bL)}{JL}s+\frac{(bR+K_1^2)}{JL}} \tag{5.4}$$

Comparing equations (5.2) and (5.4) we have the following equalities:

$$\frac{K_1}{JL} = K_2\omega^2 \tag{5.5}$$

$$\frac{(bR+K_1^2)}{JL} = \omega^2 \tag{5.6}$$

$$\frac{(JR+bL)}{JL} = 2\zeta\omega \tag{5.7}$$

From (5.5) y (5.6) he have:

$$\frac{K_1}{K_2JL} = \frac{(bR+K_1^2)}{JL} \tag{5.8}$$

Solving the equation R:

$$R = \frac{K_1}{bK_2} - \frac{K_1^2}{b} \tag{5.9}$$

Replacing in (5.9) the given values and calculating $K_2 \approx 1.987$ (value where the system stabilizes, see Figure 5.2), we have $R = 0.1636\Omega$.

Therefore, to obtain the value of the inductance, it is necessary to know the value of ζ, known as the damping coefficient, which is related to the percentage of damping, MP, according to the equation:

$$MP = 100e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}} \tag{5.10}$$

The value of MP is obtained from the plot of the system response to a step signal, as shown in Figure 5.3, which was calculated to be approximately 47%.

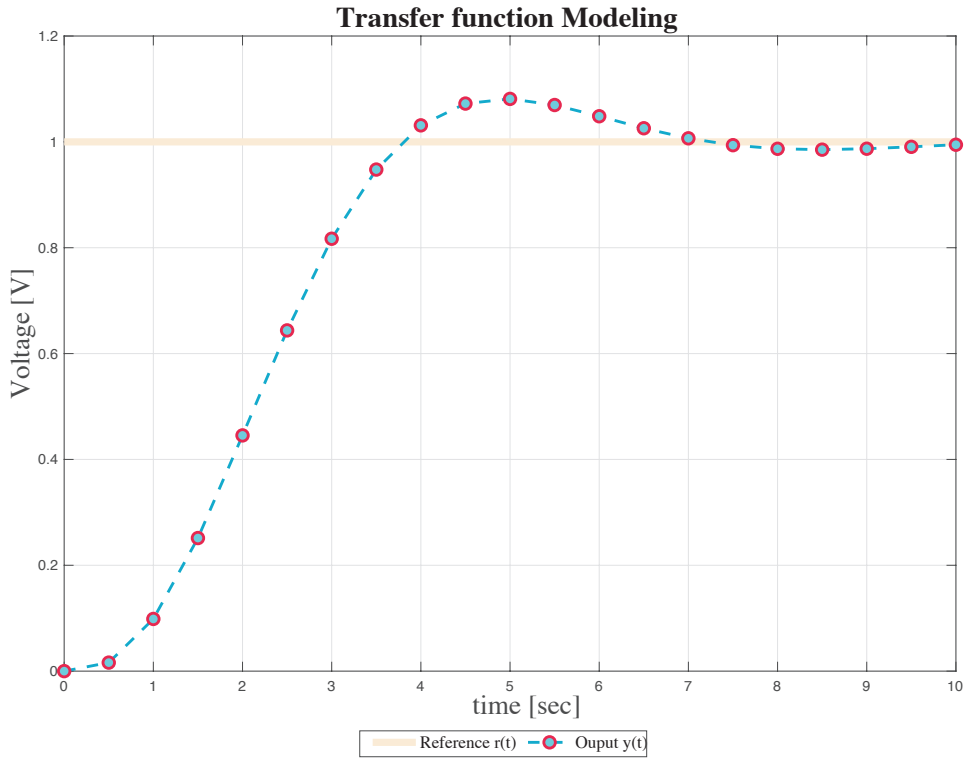


Figure 5.3: System response to a step of 1 V

Solving (5.10) from ζ, it is obtaining $\zeta = 0.2336$.

Then, solving the value of ω de (5.5), $\omega = \sqrt{\frac{K_1}{K_2 JL}}$, and replacing it in (5.7)

$$\frac{(JR + bL)}{JL} = 2\zeta\sqrt{\frac{K_1}{K_2 JL}} \tag{5.11}$$

of the equation (5.11), all values are known except L. Therefore, we will try to solve it for this variable.

Solving (5.11), with the data values of J, K, b and the calculated values of R and ζ, yields the following values of L.

$$L_1 = 0.2592H \qquad L_2 = 258.0144H \tag{5.12}$$

Figure 5.4 shows the system response for the two values of L, with L_1 being the inductance that gives the closest answer to the desired system.

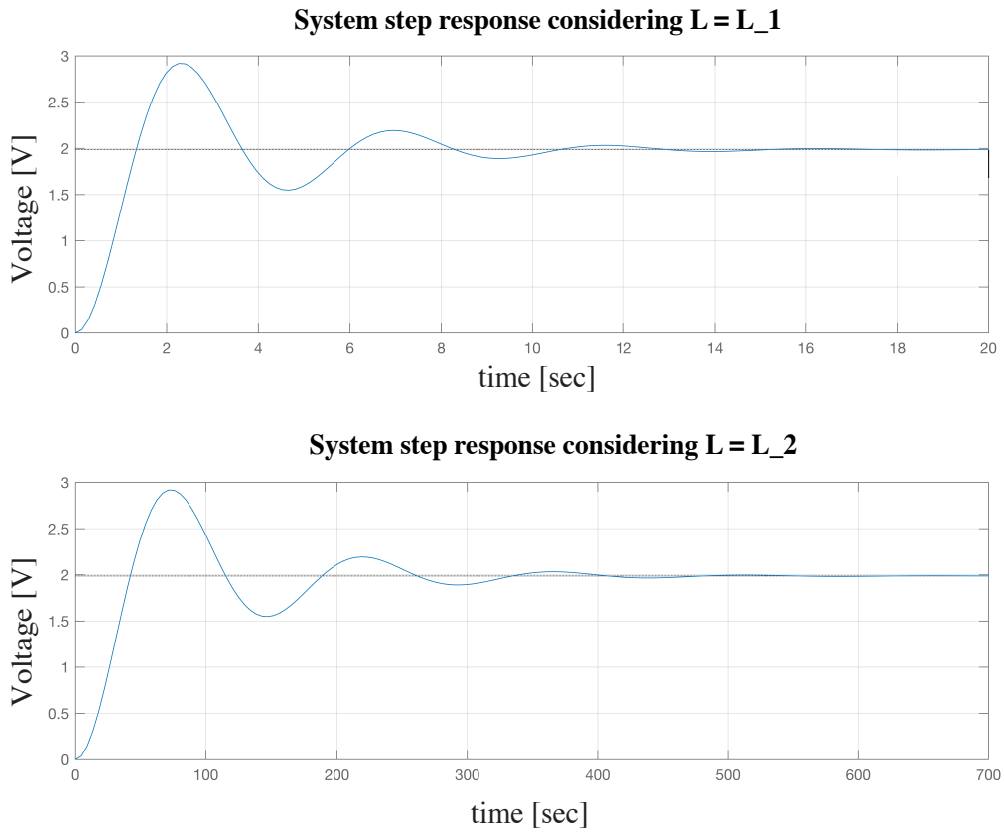


Figure 5.4: Response of the system to the step for the two values of L

Thus, the transfer function of the system is:

$$G(s) = \frac{\dot{\theta}}{V} = \frac{3.58}{s^2 + 0.651s + 1.941} \tag{5.13}$$

5.3.2 System identification with b = 0

Initially in $b = 0$ it is obtained:

$$G_1(s) = \frac{\frac{K_1}{JL}}{s^2 + \frac{R}{L}s + \frac{K_1^2}{JL}} \tag{5.14}$$

Being the equalities:

$$\frac{K_1}{JL} = K_2\omega^2 \tag{5.15}$$

$$\frac{K_1^2}{JL} = \omega^2 \tag{5.16}$$

$$\frac{R}{L} = 2\zeta\omega \tag{5.17}$$

From (5.15) y (5.16) it is obtained:

$$K_2 = \frac{1}{K_1} = 2$$

Now, from (5.16) and (5.17) we have:

$$L = \frac{JR^2}{\zeta^2} \tag{5.18}$$

Thus, we have a parametric equation, L, as a function of R; therefore, it is impossible to directly determine the values of R and L. However, if we take the value of R found in the previous point ($R = 0.16346\Omega$), we have $L = 0.2452H$, and the response of the system is similar to that in Figure 5.4, as shown in 5.5.

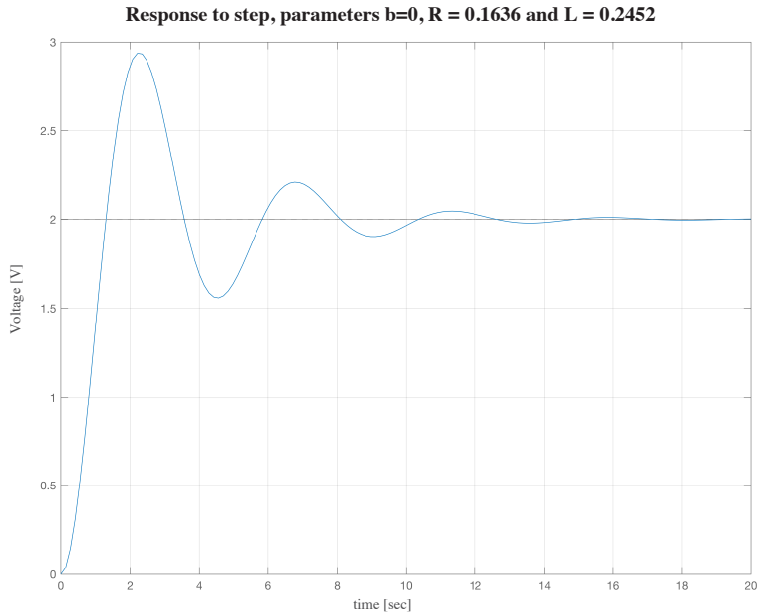


Figure 5.5: Response of the system, with $b = 0$, to the step

Thus, the transfer function of the system is:

$$G(s) = \frac{\dot{\theta}}{V} = \frac{4.078}{s^2 + 0.6671s + 2.039} \tag{5.19}$$

It is then concluded that the value of b does not influence the system too much, which was expected, since some authors ignore it to simplify the modeling of a DC motor.

5.3.3 Part two: PID control

It was decided to develop a PID controller to implement the system in Simulink. Then, to do so, the system was discretized using the command `c2d(TF, T)`, where `TF` is the system's transfer function in continuous time, and T is the sampling period. In this case, $T = 0.05$ seconds was used. Thus the transfer function is:

$$G(z) = \frac{0.004769z + 0.004717}{z^2 - 1.963z + 0.968} \tag{5.20}$$

The system, already with the PID controller, is shown in Figure 5.6.

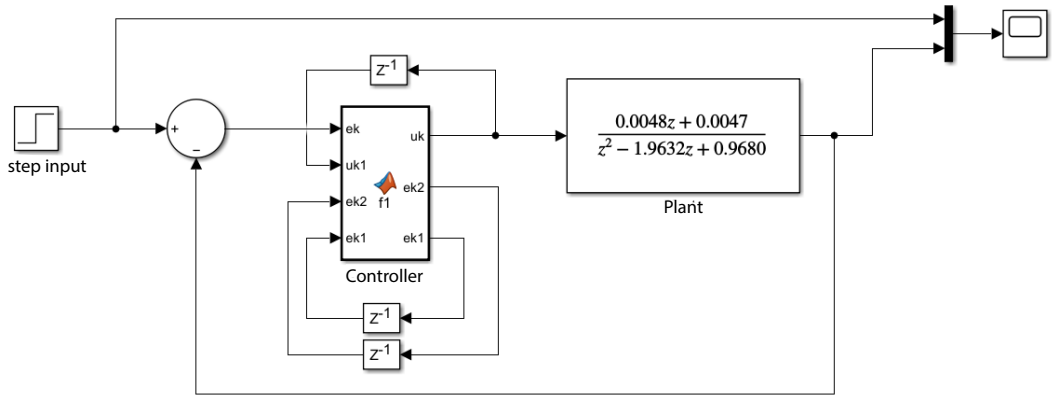


Figure 5.6: System simulation in Simulink

A MATLAB function block is used to program the PID. This block contains the following code.

```
function [uk,ek2,ek1]  = f1(ek,uk1,ek2,ek1)
    kp = 1;
    ki =  4;
    kd =  2;
    T = 0.05;
    uk = uk1 + (kp + ki*T/2 + kd/T)*ek + (-kp + ki*T/2 - 2*kd/T)*ek1 + kd/T*ek2;
    ek2 = ek1;
    ek1 = ek;
end
```

The PID calibration was manual. First, the derivative constant was set to 0 and the integral constant to a minimal value, then the proportional constant was varied until the system reached the reference. The next step was to increase the value of the integrating constant to have a PI control; however, the system became unstable, so giving a discount to the derivative constant was necessary. In this way, the best results were obtained with:

$$K_p = 1; \qquad K_i = 4; \qquad K_d = 2$$

The controller result is shown in Figure 5.7. Whose percentage of overshoot is 0.9743%.

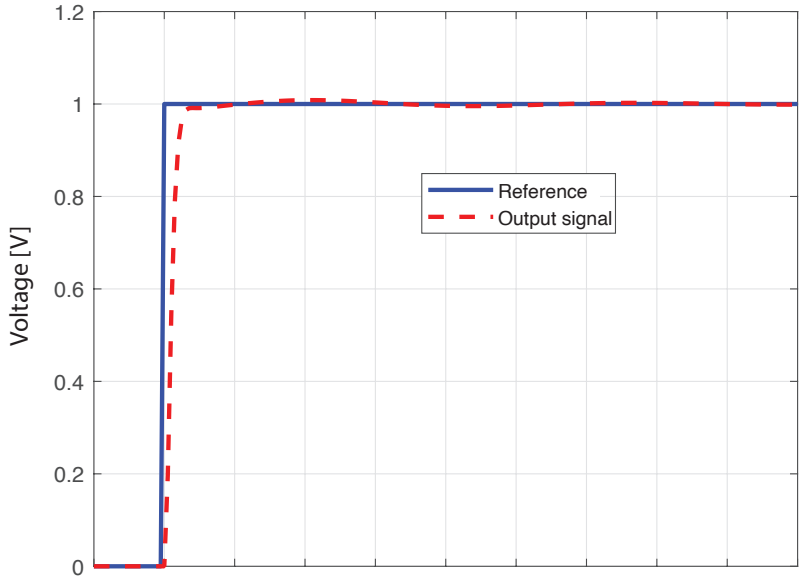


Figure 5.7: System response with PID controller

Figure 5.8 shows the system's response to reference changes. It is clear that the system quickly follows the reference.

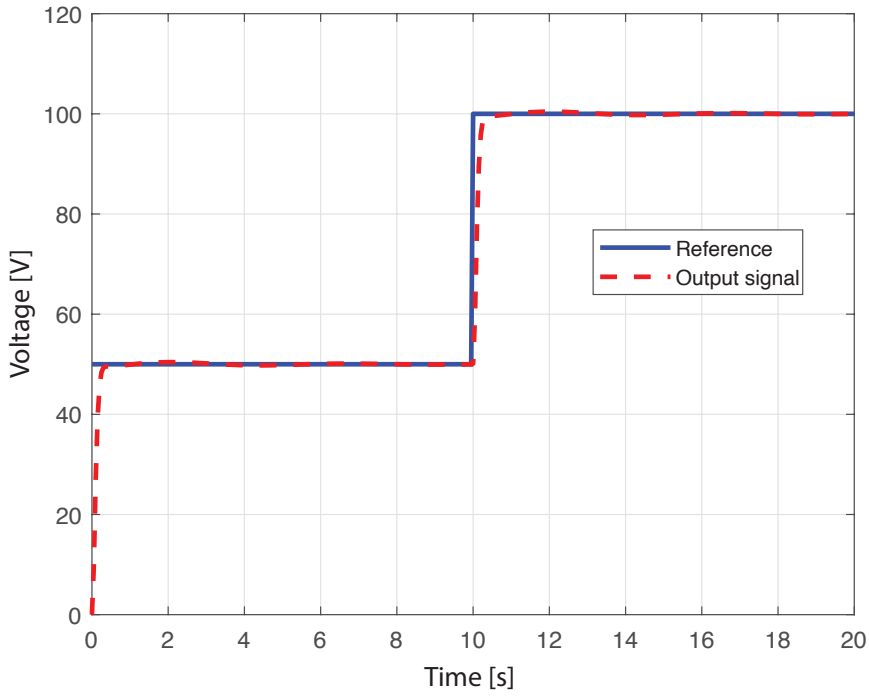


Figure 5.8: Response of the system with PID controller to reference changes

5.3.4 Part three: feedback control

For this part, a control known as an integrating follower was implemented, whose control scheme is shown in Fig. 5.9.

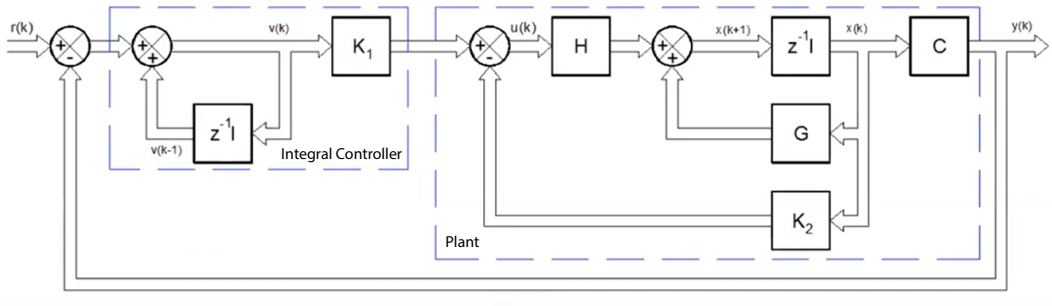


Figure 5.9: Tracking control system with integrator

Being the control law:

$$u(k) = -K_2 x(k) + K_1 v(k) \tag{5.21}$$

This type of controller raises the degree of the system by 1. In this case, the system is of order 2, which will become of order 3 when using the integrating follower.

The controller gains K_1 and K_2 are calculated using Ackerman's formula and Bessel polynomials to bring the poles to a specific location, depending on the desired settling time.

In this case, having seen the response of the PID controller, a settling time of 0.1 seconds is assumed, and the Bessel polynomials of degree 3, taken from Figure 5.1, are used.

Table 5.1: Variables of Roots of Bessel polynomials

Polynomial order	Roots
1	-4.6200
2	$-4.0530 \pm 2.34j$
3	$-5.0093, -3.9669 \pm 3.7845j$
4	$-4.0156 \pm 5.0723j, -5.5281 \pm 1.653j$
5	$-6.4480, -4.1104 \pm 6.3142j, -5.9268 \pm 3.0813j$
6	$-4.2169 \pm 7.53j, -6.2613 \pm 4.4018j, -7.1205 \pm 1.4540j,$
7	$-8.0271, -4.3361 \pm 8.7519j, -6.5714 \pm 5.6786j, -7.6824 \pm 2.8081j$

To obtain the state-space system, the command `tf2ss(num, den)` is used, which returns the state-space matrices of the transfer function, whose numerator and denominator are `num` and `den`, respectively.

The system's response is shown in Figure 5.10, where it is observed how fast the plant follows the reference (settling time of 0.1 seconds); there is an overshoot of 0.6425%.

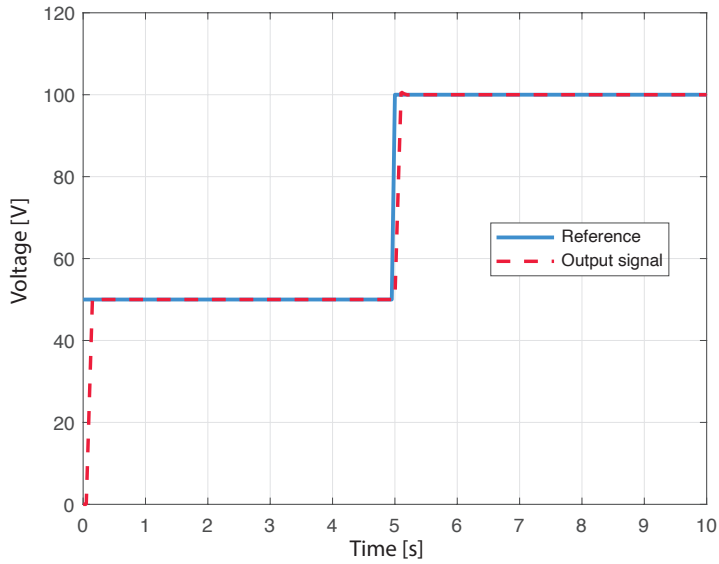


Figure 5.10: System response with integrating follower controller

As a final part, an MPC(Model predictive control) is proposed for the speed control of a DC motor. It is an advanced process control method used by the process industry in chemical plants and oil refineries since the 1980s. In recent years it has also been used in stability models for power systems. MPC can anticipate future events and take control actions accordingly.

This control scheme requires the system in state space to calculate gains that will be used to determine the control variable. The control law for this scheme is:

$$u(k+1) = u(k) + \Delta U(k+1)$$

Being the equations governing the system:

$$Y = Fx(k_i) + \Phi \Delta U$$

$$F = \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^{N_p} \end{bmatrix}$$

$$\Phi = \begin{bmatrix} CB & 0 & 0 & \cdots & 0 \\ CAB & CB & 0 & \cdots & 0 \\ C^2AB & CAB & CB & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{N_p-1}B & CA^{N_p-2}B & A^{N_p-3}B & \cdots & CA^{N_p-N_c}B \end{bmatrix}$$

After formulating the mathematical model, the next step in designing an MPC control system is calculating the predicted plant output with the future control

signal as the adjustable variable. This prediction is described within an optimization window, with the length of the optimization window being the variable Np (the number of samples).

The future control path is denoted by:

$$\Delta u(k_i), \Delta u(k_i + 1), \ldots, \Delta u(k_i + N_c - 1) \tag{5.22}$$

Where N_c is called the control horizon, which determines the number of parameters used to calculate the future trajectory of the system, there is also the parameter r_w, which is used as a tuning parameter for the controller performance.

5.4 Results and Analysis

The system's response with the MPC controller is shown below, where it is clear that it has a settling time of about 0.3 seconds, and there is no overshoot. Figure 5.11 shows the system's behavior to changes in the reference.

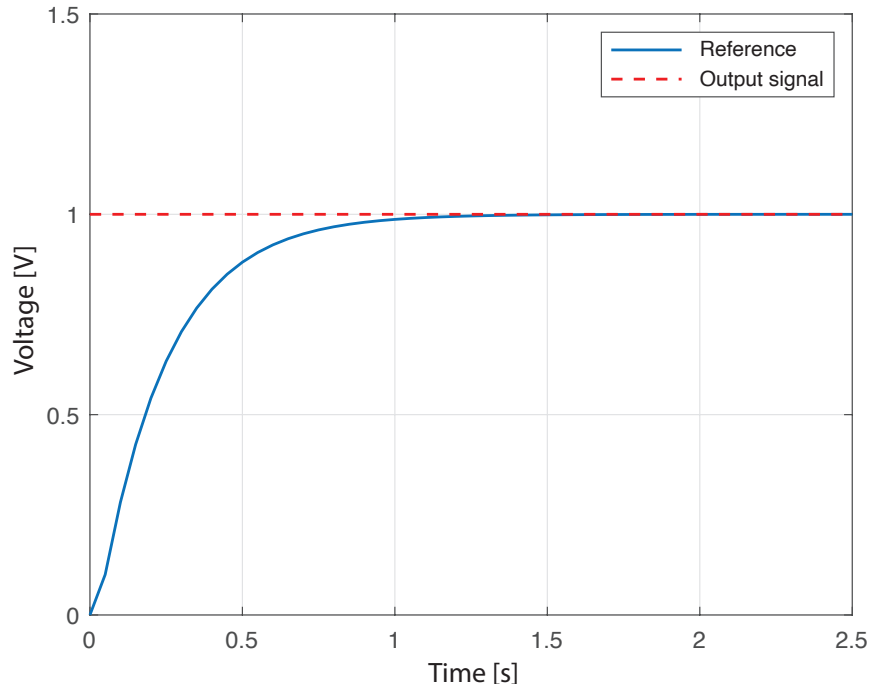


Figure 5.11: System response with MPC controller

It is observed in Figure 5.11 that the system is somewhat slow, almost 1 second in reaching the reference; this is because, although it does not occur here, in Simulink, an over impulse was presented, and if the system with a faster response was made, the over impulse grew to more than 10%.

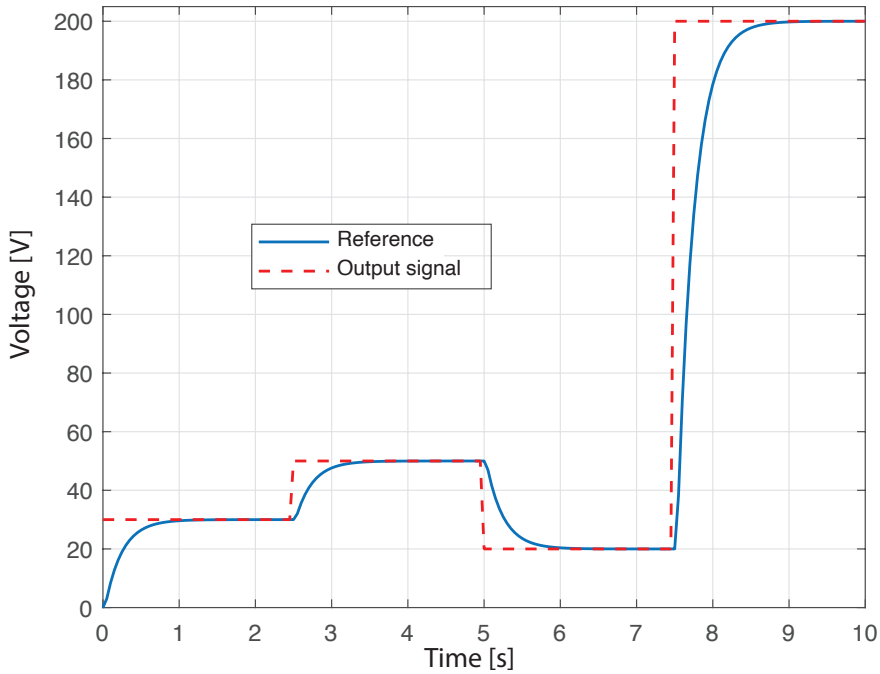


Figure 5.12: Response of the system with MPC controller to reference changes

As said previously, in Figure 5.12, there is no overshoot; when simulating the system in Simulink, there was one, as shown in Figure 5.13. It is approximately 3.6%.

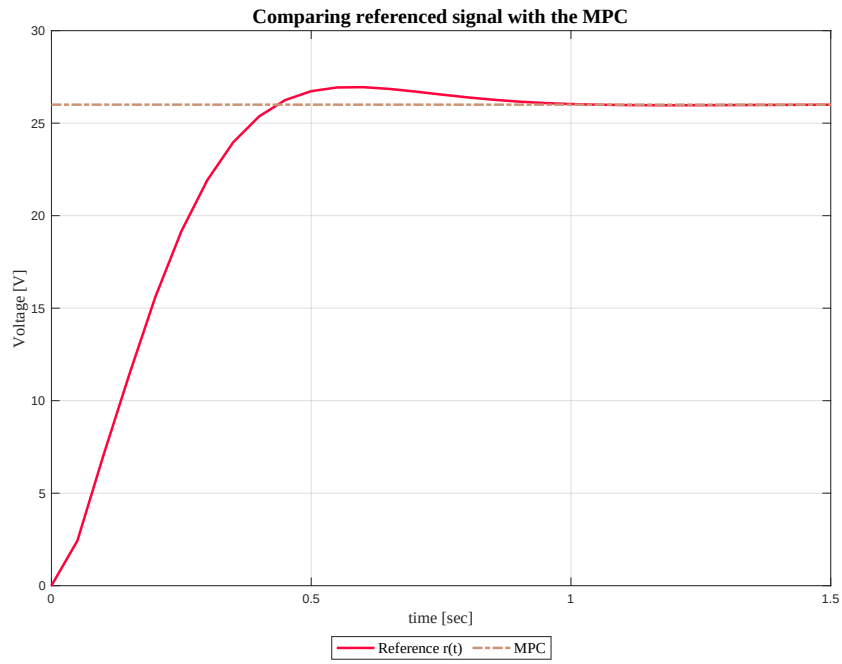


Figure 5.13: System response to a 1V step, in Simulink

The table 5.2 indicates the main results of the controllers. Here it is observed that the integrating follower controller shows the best performance, followed by the PID and finally the MPC.

Table 5.2: Controller results

	PID	**Integrator**	**MPC**
Settling time (s)	0.5	0.1	1
% of the maximum peak	0.9743	0.6425	3.6

Finally, the three controllers are shown in Figure 5.14. It can be seen how the integrator is the first to reach the reference, followed by the PID and then the MPC, with a small pulse sip. The PID shows some oscillation around the reference but quickly stabilizes. In conclusion, all three controllers manage to reach the target.

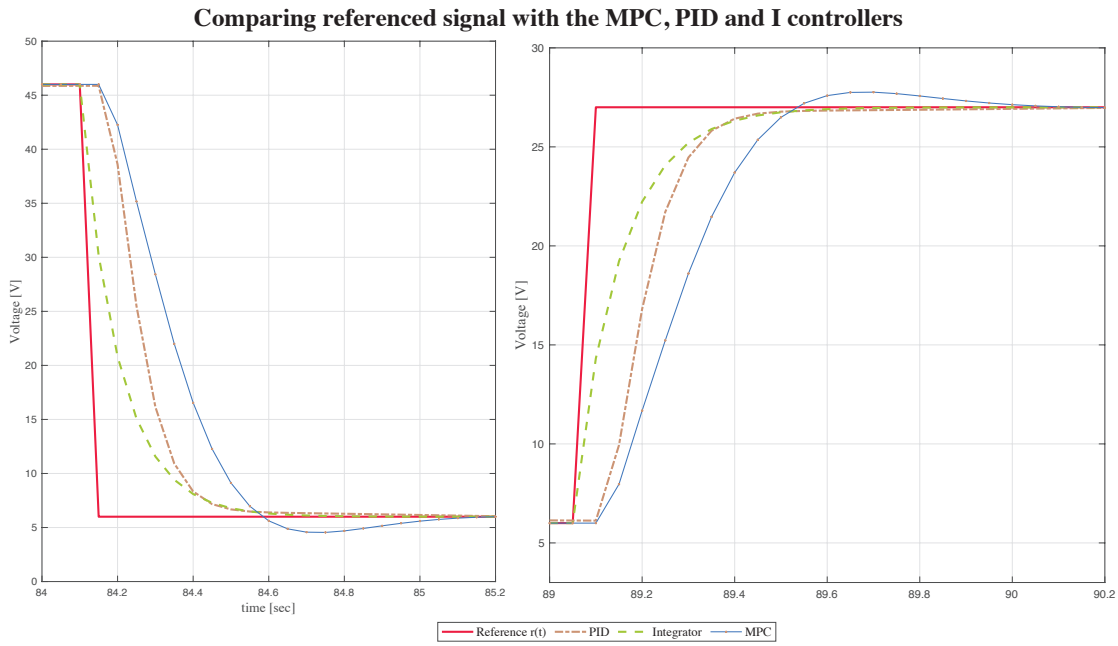


Figure 5.14: Controller response to reference changes

5.5 Conclusions

A practical model for the Direct Current (DC) motor could be developed by utilizing the available data and analyzing the time response graph. This model accurately represents the behavior of the motor and provides valuable insights into its performance. The successful modeling of the DC motor enhances our understanding of its dynamics and facilitates further analysis and optimization in various applications.

Three distinct controllers were developed for the DC motor system through the design process. Each controller exhibited varying performance characteristics, demonstrating suitability for specific application requirements. This achievement

highlights the versatility and adaptability of the control system, enabling optimization based on desired outcomes. The availability of multiple controller options enhances flexibility in addressing diverse operational needs and paves the way for efficient and effective DC motor control in various contexts.

The Model Predictive Controller (MPC) exhibited relatively poorer performance than the other two controllers. However, this does not imply its inefficiency, as further testing is required to ascertain its optimal performance. Therefore, it is recommended to implement the MPC controller on an entire system to evaluate its effectiveness and determine its true potential. This additional experimentation will provide valuable insights and contribute to the ongoing development of efficient control strategies for practical applications.

In summary, fractional order proportional-integral-derivative (FOPID) controllers replace traditional proportional-integral-derivative (PIDs) in real-time control systems. FOPID controllers provide more degrees of freedom than traditional PIDs since their architecture has fractional-order components. This capability, which classical PIDs lack, enables improved controller parameter adjustment, improving closed-loop performance or transient responsiveness when the controllers are applied to nonlinear systems.

Bibliography

[1] A. Tepljakov, B. B. Alagoz, C. Yeroglu, E. Gonzalez, S. H. HosseinNia, and E. Petlenkov, "FOPID Controllers and Their Industrial Applications: A Survey of Recent Results11This study is based upon works from COST Action CA15225, a network supported by COST (European Cooperation in Science and Technology).", *IFAC-PapersOnLine*, vol. 51, no. 4, pp. 25–30, 2018, ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2018.06.014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896318303045.

[2] I. Podlubny, "Fractional-order systems and PI/sup /spl lambda//D/sup /spl mu//-controllers", *IEEE Transactions on Automatic Control*, vol. 44, no. 1, pp. 208–214, 1999, ISSN: 1558-2523 VO - 44. DOI: 10.1109/9.739144.

[3] A. Fekik, H. Denoun, A. T. Azar, *et al.*, "Adapted Fuzzy Fractional Order proportional-integral controller for DC Motor", in *2020 First International Conference of Smart Systems and Emerging Technologies (SMARTTECH)*, 2020, pp. 1–6, ISBN: 978-1-7281-7407-5. DOI: 10.1109/SMART-TECH49988.2020.00019.

[4] A. Al-Mayyahi, W. Wang, and P. Birch, "Path tracking of autonomous ground vehicle based on fractional order PID controller optimized by PSO", in *2015 IEEE 13th International Symposium on Applied Machine Intelligence and Informatics (SAMI)*, 2015, pp. 109–114, ISBN: 978-1-4799-8221-9. DOI: 10.1109/SAMI.2015.7061857.

[5] R. Sadeghian and M. T. Masoule, "An experimental study on the PID and Fuzzy-PID controllers on a designed two-wheeled self-balancing autonomous robot", in *2016 4th International Conference on Control, Instrumentation, and Automation (ICCIA)*, 2016, pp. 313–318, ISBN: 978-1-4673-8704-0. DOI: 10.1109/ICCIAutom.2016.7483180.

[6] S. Pinzón and W. Pavón, "Diseño de Sistemas de Control Basados en el Análisis del Dominio en Frecuencia", *Revista Técnica "Energía"*, vol. 15, no. 2, pp. 76–82, 2019, ISSN: 1390-5074. DOI: 10.37116/revistaenergia.v15.n2.2019.380.

[7] J. Aguila-Leon, C. Vargas-Salgado, C. Chiñas-Palacios, and D. Díaz-Bello, "Solar photovoltaic Maximum Power Point Tracking controller optimization using Grey Wolf Optimizer: A performance comparison between bio-inspired and traditional algorithms", *Expert Systems with Applications*, vol. 211, no. May 2022, 2023, ISSN: 09574174. DOI: 10.1016/j.eswa.2022.118700.

[8] H. Li, Z. Ren, M. Fan, *et al.*, "A review of scenario analysis methods in planning and operation of modern power systems: Methodologies, applications, and challenges", *Electric Power Systems Research*, vol. 205, no. November 2021, p. 107 722, 2022, ISSN: 03787796. DOI: 10.1016/j.epsr.2021.107722. [Online]. Available: https://doi.org/10.1016/j.epsr.2021.107722.

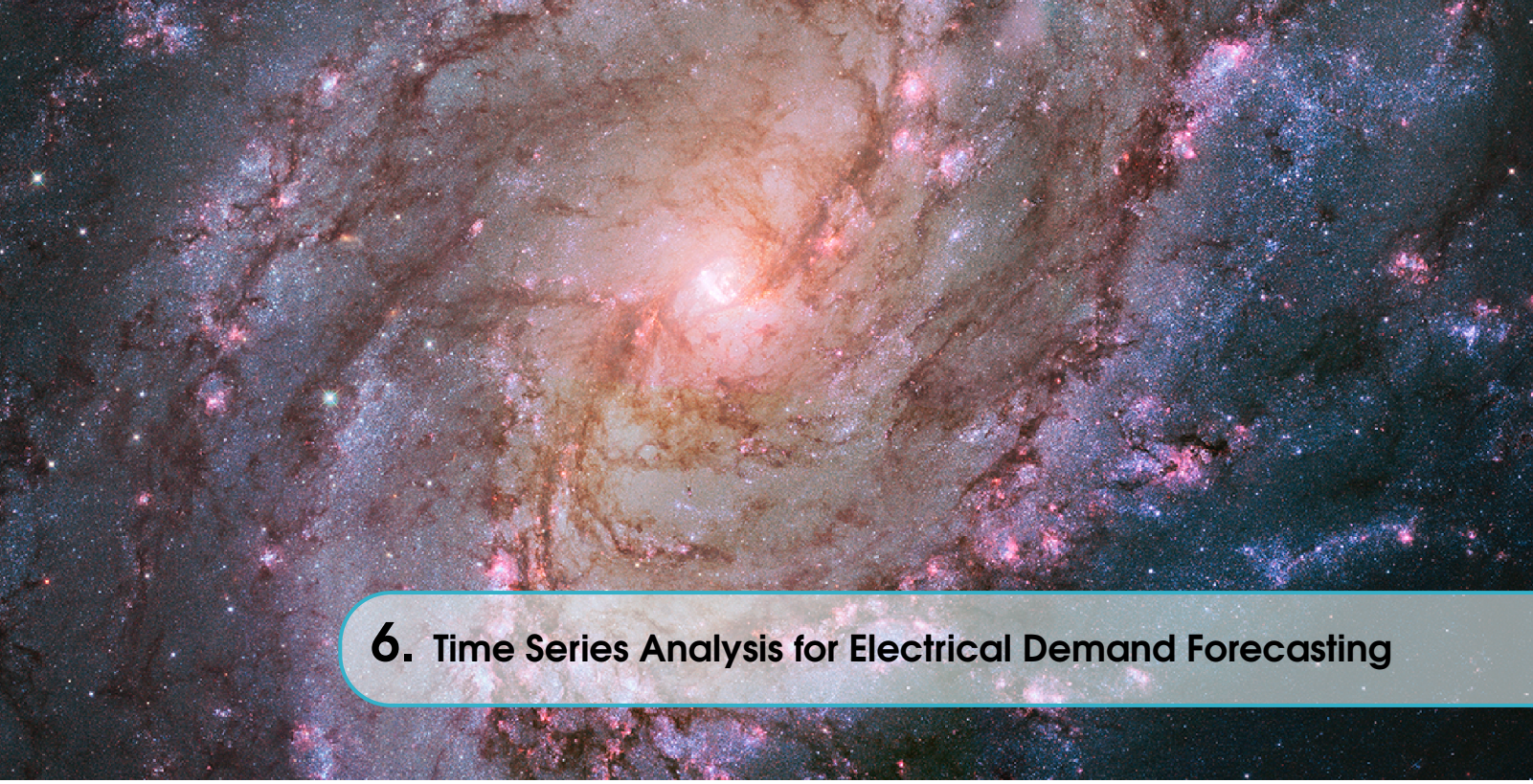
[9] F. d. J. Sorcia-Vázquez, J. Y. Rumbo-Morales, J. A. Brizuela-Mendoza, *et al.*, *Experimental Validation of Fractional PID Controllers Applied to a Two-Tank System*, 2023. DOI: 10.3390/math11122651.

[10] C. Zhang, K. Peng, L. Guo, C. Xiao, X. Zhang, and Z. Zhao, "An EVs charging guiding strategy for the coupling system of road network and distribution network based on the PT3", *Electric Power Systems Research*, vol. 214, no. PA, p. 108 839, 2023, ISSN: 03787796. DOI: 10.1016/j.epsr.2022.108839. [Online]. Available: https://doi.org/10.1016/j.epsr.2022.108839.

[11] V. T. Aghaei, A. Seyyedabbasi, J. Rasheed, and A. M. Abu-Mahfouz, "Sand Cat Swarm Optimization-Based Feedback Controller Design for Nonlinear Systems", *SSRN Electronic Journal*, vol. 9, no. 3, e13885, 2023, ISSN: 2405-8440. DOI: 10.2139/ssrn.4312627. [Online]. Available: https://doi.org/10.1016/j.heliyon.2023.e13885.

[12] Y. Song, X. Cai, X. Zhou, *et al.*, "Dynamic hybrid mechanism-based differential evolution algorithm and its application", *Expert Systems with Applications*, vol. 213, no. PA, p. 118 834, 2023, ISSN: 09574174. DOI: 10.1016/j.eswa.2022.118834. [Online]. Available: https://doi.org/10.1016/j.eswa.2022.118834.

[13] S. Tufenkci, B. Baykant Alagoz, G. Kavuran, C. Yeroglu, N. Herencsar, and S. Mahata, "A theoretical demonstration for reinforcement learning of PI control dynamics for optimal speed control of DC motors by using Twin Delay Deep Deterministic Policy Gradient Algorithm", *Expert Systems with Applications*, vol. 213, no. PC, p. 119 192, 2023, ISSN: 09574174. DOI: 10.1016/j.eswa.2022.119192. [Online]. Available: https://doi.org/10.1016/j.eswa.2022.119192.



6. Time Series Analysis for Electrical Demand Forecasting

M.Sc. Manuel Jaramillo Monge is a Professor of the Electrical Engineering Career at the Universidad Politécnica Salesiana (Ecuador)
Email: mjaramillo@ups.edu.ec
https://orcid.org/0000-0002-1714-222X
DOI: 10.17163/abyaups.44.351

6.1 Introduction

Accurate load forecasting is vital for efficient energy management and planning in smart grids. In the revision of the literature review, it can be seen the principal authors that have been published regarding energy demand forecast are shown in figure 6.1; also, in figure 6.2, the major countries that have contributed to this research topic are shown.

Energy forecasting enables power system operators to manage energy supply and demand, minimize energy waste, and maintain the regular operation of power systems. Load forecasting also plays a critical role in integrating renewable energy sources into the power grid, as it helps balance the energy supply and demand and reduces dependence on fossil fuels (1).

Accurate load forecasting is crucial for managing residential, commercial, and industrial energy usage. It allows energy providers to plan and optimize their energy generation, transmission, and distribution systems, resulting in significant cost savings and improved energy efficiency. Moreover, load forecasting ensures that energy providers meet customers' demands while maintaining a stable and reliable supply. In all sectors, and especially industrial ones, where economic factors are critical, forecasting is particularly crucial due to their complex energy consumption patterns that vary depending on the time of day, day of the week, and academic calendar (2).

Time series analysis is a valuable statistical technique that facilitates analyzing and predicting trends over time. In energy demand forecasting, time series analysis enables the identification of patterns within energy consumption data, including daily, weekly, and seasonal variations. With this information, predictive models can be created to aid energy providers in forecasting future energy demand and designing effective plans. Using time series analysis can lead to more efficient and effective energy management.

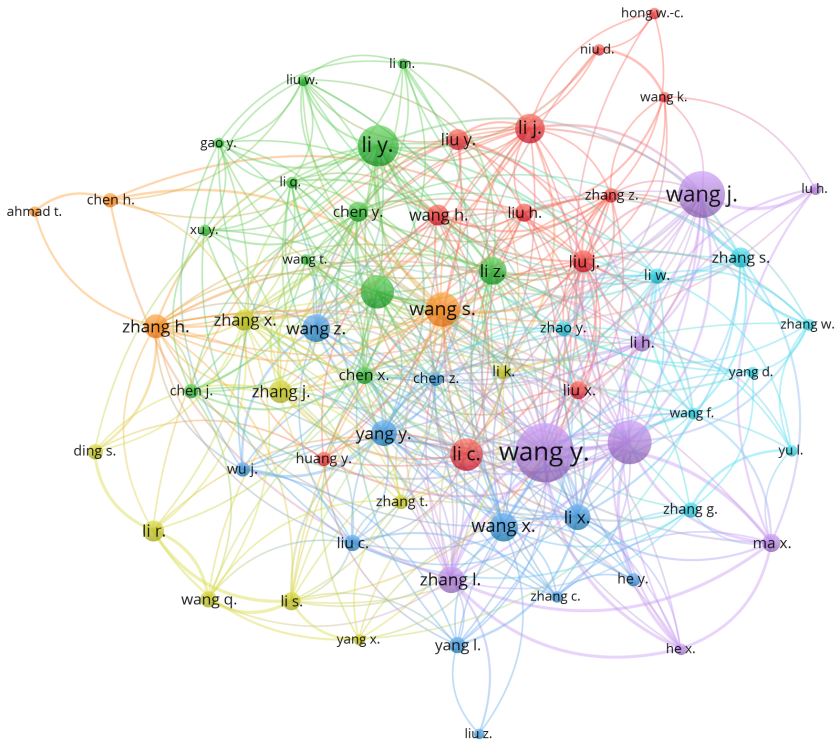


Figure 6.1: Bibliometric revision. Main authors that have published for forecasting electrical demand

6.2 Related Works

In (2), authors aim to enhance the accuracy of short-term load forecasting in smart grids by analyzing user behavior patterns and grouping users using the K-means algorithm. The research suggests a load forecasting model that employs the FCM-BP neural network to predict the load value by determining the load rate-of-change just before the prediction without making a direct prediction. The study gathered load data from users, grouped them with the K-means algorithm, and developed a load forecasting model using the FCM-BP neural network. The outcomes indicate that the proposed method improves the model's prediction accuracy compared to other methods such as RBF, GRNN, BP neural network, and the unclustered FCM-BP model. The paper also discusses future work to improve

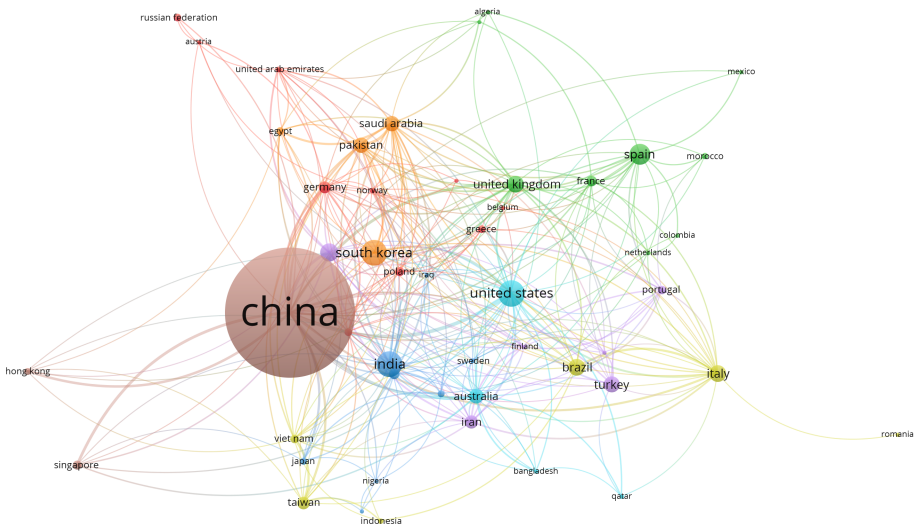


Figure 6.2: Bibliometric revision. Main countries that have published for forecasting electrical demand

the clustering method further and investigate the effect of economic and climatic factors on the accuracy of the load forecasting model. This research contributes to the field of load forecasting by suggesting an approach that enhances the accuracy of short-term load forecasting in smart grids. The proposed method leverages user behavior patterns and grouping to boost the accuracy of user-level load forecasting, emphasizing the importance of analyzing user behavior patterns and clustering users to enhance load forecasting accuracy. Power system operators can apply the proposed method to improve load forecasting accuracy, aiding energy management and planning. The research also recommends future work to enhance the clustering method further and analyze the impact of external factors, such as economic and climatic factors, on the accuracy of the load forecasting model. The proposed method can construct a user-level high-resolution and high-precision load forecasting model.

Authors in (3) present power consumption forecasting models utilizing artificial neural networks (ANN) and support vector regression (SVR) for higher education institutions. The study collected power consumption data every 15 minutes for over a year from four different building clusters in a university. The research team evaluated three other data models that represented the building clusters' characteristics and assessed performance using mean absolute percentage error (MAPE), root mean square error (RMSE), and mean fundamental error (MAE). Results showed that the ANN-based model outperformed the SVR-based model. Principal component analysis (PCA) was also more accurate than factor analysis (FA).

Additionally, including work hours, class schedule information, and weather data improved forecasting accuracy, particularly for educational buildings. These models can aid higher education institutions in optimizing energy management strategies and achieving sustainability goals. The research team compared their work with previous studies using machine learning algorithms such as ANN, GA-

ANN, and SVR for power consumption forecasting. Their models predicted energy consumption every 15 minutes, while previous studies predicted it hourly. They found that considering external factors such as weekday/holiday and weather information improved prediction accuracy. The team evaluated their models' performance using MAPE, RMSE, and MAE, demonstrating that their proposed models achieved higher accuracy than prior studies. The team also presented prediction results for events and average days using the best forecasting model for each cluster. The experimental results indicate that the models can accurately predict power consumption for higher education institutions, resulting in significant cost savings, reduced carbon emissions, and improved energy efficiency.

In (4), the paper aims to create a mathematical model that can accurately forecast the amount of active power consumed in the power system of Kaliningrad oblast. It involves examining the structure of electric power consumption, studying the influence of meteorological factors on active power consumption, and developing a reliable mathematical model for operational forecasting. The study uses statistical research methods, specifically regression analysis, to determine how meteorological factors impact active power consumption. The developed mathematical model is evaluated using six-time series forecasting error indicators, and its reliability is confirmed using the Fisher criterion. The results demonstrate that the mathematical model is highly accurate in forecasting active power consumption, with an error probability of less than 1%. The study's novelty lies in developing a mathematical model incorporating the adjusted change rate of dynamic power consumption according to discrete conditions and the amount of active power used for power plant needs. The practical significance of this model is that it serves as the basis for the method of operational forecasting, which is implemented in the Regional Dispatch Department of the Kaliningrad Oblast's Power System. The study's findings can be helpful for power system operators and researchers to develop similar mathematical models to forecast active power consumption in other power systems, as the developed model accurately predicts active power consumption with a forecasting error of less than 3.0% and an error probability of less than 1%.

The research presented in (5) proposes a more accurate and faster residential electricity consumption forecasting model called the IWOA-OGSVI model. It combines the Improved Whale Optimization Algorithm (IWOA) and the Optimized Grey Seasonal Variation Index (OGSVI) model and outperforms traditional models in accuracy and speed. The paper includes an empirical study on residential electricity consumption in four Chinese cities and suggests future research directions. Overall, the article provides valuable insights into developing accurate and prompt forecasting models for electricity consumption.

In (6), the author explores the potential of forecasting electricity consumption for a large healthcare facility. The forecasting technique used is SARIMA modeling, which analyzes past data to predict future values. The study employs the Box-Jenkins procedure to create possible models and compares the performance of SARIMA and ARIMA models. The findings indicate that the SARIMA model outperforms the ARIMA model. The analysis of hospital data spanning 11 years reveals that these dynamic models can accurately forecast electricity consumption.

Research in (7) aims to determine the most reliable short-term load forecasting

method for residential consumers during unusual consumption, such as the COVID-19 pandemic. The study evaluates three forecasting methods: linear regression (LR), autoregressive integrated moving average (ARIMA), and artificial neural network (ANN). The authors used multiyear hourly residential consumption data to estimate and validate the accuracy of the forecasts. The main findings indicate that the forecasting methods retained their hierarchy and accuracy in forecasting errors during unusual consumer behavior, similar to normal conditions if a trigger or alarm mechanism existed and there was sufficient time to adapt and deploy the forecasting algorithm. The ANN method generated the best results, followed by ARIMA and LR. The paper's original contribution is the ability to forecast loads that have no historical reference data. The results can be used as best practices during power load uncertainty and unusual consumption behavior.

Authors in (8) focus on accurately forecasting India's reliance on foreign oil using different methods. The authors utilized linear Auto-Regressive Integrated Moving Average (ARIMA) and nonlinear Back Propagation (BP) to correct the nonlinear metabolic grey model (NMGM) forecasting residuals in three steps. They integrated the metabolic idea with a nonlinear grey model to create NMGM, which they combined with ARIMA to develop NMGM-ARIMA and BP to develop NMGM-BP. The proposed models analyzed India's dependence on foreign oil from 1995 to 2017 and forecasted the data from 2018 to 2030. The mean relative error of the proposed forecasting models was approximately 1.5%, which produced reliable results. The forecasts demonstrate that India's dependence on foreign oil is expected to increase to 90% around 2025, which poses a significant challenge to India's oil security and global oil market.

Research in (9) explores how well ARIMA models can predict electrical load time series despite noise interference. They conducted a simulation-based experiment using an ARIMA model based on actual load data from the Polish power system. They introduced varying levels of noise to test its forecast accuracy. The model was then re-identified, its parameters estimated, and new forecasts made to determine the threshold at which the model's forecasting ability broke down. These findings underscore the significance of data preprocessing in data mining and learning and have possible implications for energy policy and power system reliability.

Other techniques for forecasting are also widely used, as shown in (10), where researchers present a model for predicting power consumption that uses wavelet transform and multi-layer LSTM. The goal is to create an effective power generation and transmission plan while avoiding the waste of electricity resources. The sample data is first processed to eliminate any volatility in the electricity consumption data using wavelet transform to achieve this. Then, the multi-layer LSTM model is used for training using the pre-processed samples. The model is tested by predicting daily power consumption in an area controlled by a US electric power company. Experimental results show that this model has better prediction performance than traditional LSTM and bidirectional LSTM, with a mean square error of 0.019 and a coefficient of determination R2 as high as 0.997. Additionally, wavelet denoising can further improve the prediction performance of the model.

In (11), the paper presents a new way of predicting monthly electricity usage using a hybrid forecasting model. The model combines the HolteWinters exponential

smoothing method with the fruit fly optimization algorithm to accurately forecast occasional series, even with limited training data. The fruit fly optimization algorithm is used to identify the best smoothing parameters for the Holt-Winters exponential smoothing. The model was tested using electricity consumption data from a city in China and was found to improve prediction accuracy, even with limited training samples significantly. Additionally, the proposed model is faster than other hybrid benchmark algorithms.

6.3 Problem Formulation and Methodology

Accurate energy forecasting is crucial for power system expansion and reliability, as it enables utilities to plan for future energy demands and ensure enough capacity to meet those demands. It is especially significant for renewable energy sources, which can be more variable than traditional fossil fuels. Utilities can better manage their energy supply and avoid blackouts or brownouts through precise energy forecasting of wind turbines or solar panels.

Also, energy forecasting is vital for preparing for future energy needs and ensuring the reliability of the power system. Utilities can use energy supply and demand predictions to identify potential weaknesses in the design and take steps to prevent them. For instance, if a utility expects high energy demand during a specific time, it may activate additional generators or introduce demand response programs to lessen energy usage during peak hours. This proactive approach guarantees that the power system remains dependable and robust, despite changing energy demands and potential disturbances (12).

Load increments uncertainties can significantly impact the stability and reliability of electrical power systems (13). Accurate load forecasting is essential for energy companies to efficiently plan their production and distribution of energy, which can help to avoid wastage or shortages. Techniques for load forecasting are essential to mitigate these uncertainties, as they can help energy companies to make more accurate predictions of future energy demand. This can lead to better resource allocation, reduced operational costs, and improved customer satisfaction. Additionally, load forecasting techniques can help energy companies to identify opportunities for energy conservation and efficiency improvements, which can further reduce the impact of load increments uncertainties on the power system.

Therefore, this work proposes using ARIMA time series for electricity consumption modeling and prediction through Matlab.

6.3.1 Study case

Therefore, to conduct a forecasting analysis using time series analysis, actual data on electricity consumption has been collected. Specifically, this work uses monthly electricity consumption data from Quito, Ecuador, from 2004 to 2018, resulting in 180 data records. This information is publicly accessible online through the official Electricity regulator entity of the Ecuadorian government. Figure 6.3 shows the electricity consumption during the period previously stated; in other words, it offers

180 monthly data; additionally, for a better understanding, Figure 6.4 shows only a year (2010) for a better understanding of the electricity consumption behavior.

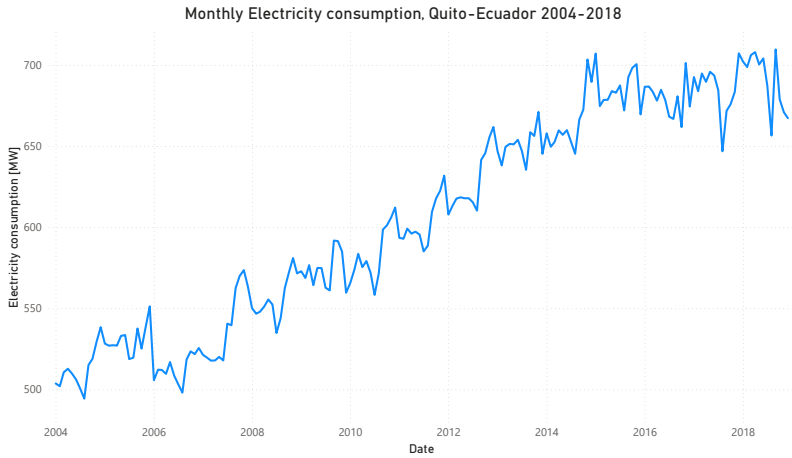


Figure 6.3: Electricity consumption for Quito-Ecuador from 2004 to 2018

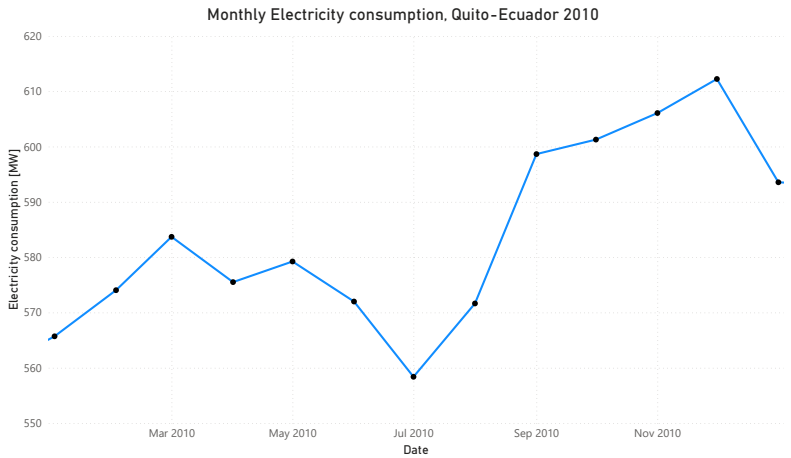


Figure 6.4: Electricity consumption for Quito-Ecuador, for the year 2010

6.3.2 Time series analysis: Seasonal Auto-Regressive Integrated Moving Average (SARIMA)

To analyze the seasonal patterns in electricity consumption, this model employs auto-regressive AR and moving average MA models. These patterns tend to repeat themselves periodically, and for this particular study, the period is 12 months based on monthly data. As a result, a SARIMA time series model would be the most appropriate for analyzing the data.

When working with autoregressive models, it is essential to remember that the current value of a given series can be expressed as a linear combination of its preceding values. This relationship holds for up to a maximum of p previous values, where p is a predetermined value dependent on the specific model being used $\{X(t-1), X(t-2), X(t-3), \ldots X(t-p)\}$. By understanding this fundamental principle, one can understand the underlying patterns and trends within a given dataset, leading to a more accurate and informative analysis. Similarly, the MA component is derived using a linear combination of q preceding white noise values $\{Z(t-1), Z(t-2), Z(t-3), \ldots Z(t-q)\}$. This method helps to smooth out data patterns and identify underlying trends.

By considering the operator B (Backward shift operator), which describes a temporal item that depends on a previous sample $BX_t = X_{t-1}$. This operator allows defining the autoregressive component of a time series as $\left\{(1 - \Phi_1 B^{12} - \Phi_2 B^{24} \ldots)X_t\right\}$ and the moving average part $\left\{(1 - \Theta_1 B^{12} - \Theta_2 B^{24} \ldots)Z_t\right\}$.

A $SARIMA$ model is made up of coefficients that determine its order. Two coefficients for the autoregressive and moving average components are denoted as p and q, respectively. Additionally, two more coefficients, represented as P and Q, are considered for the seasonal autoregressive and moving average components. The differential part is typically set to 1 for an electricity model, and the seasonality corresponds to twelve months. A $SARIMA$ model has six coefficients of interest: three for the non-periodic part (p, d, q) and three for the periodic part (P, D, Q). This information can be represented in equation (6.1).

$$\Phi_P(B^S)\phi_p(B)(1 - B^S)^D(1 - B)^d X_t = \Theta_Q(B^S)\beta_q(B)Z_t \tag{6.1}$$

6.3.3 Methodology for electricity consumption forecasting

This project aims to create a methodology for predicting electricity consumption data every month. Then, to achieve this, the data must be analyzed to determine if it displays any seasonal patterns, which is necessary for a SARIMA time series model. The range for coefficients (p,q) and (P, Q) will be identified through correlation and autocorrelation analysis. Using Matlab, the values for each coefficient will be calculated to establish the model. The final step involves forecasting the data and evaluating the error. A detailed outline of this process can be found in Figure 6.5

When the $SARIMA$ model is created, it is necessary to validate the model's error by comparing the original data for electricity consumption EC and the values produced by the model. The residual sum of squares will be used as the criteria for model validation, as is shown in equation 6.2.

$$RSS = \sum_{i=1}^{n} (ED_i - SARIMA_i)^2 \tag{6.2}$$

6.4 Matlab Coding and Results Analysis

Thus, the primary action is to meticulously clean and validate the data to initiate the suggested approach. It is essential to ascertain whether the data manifests

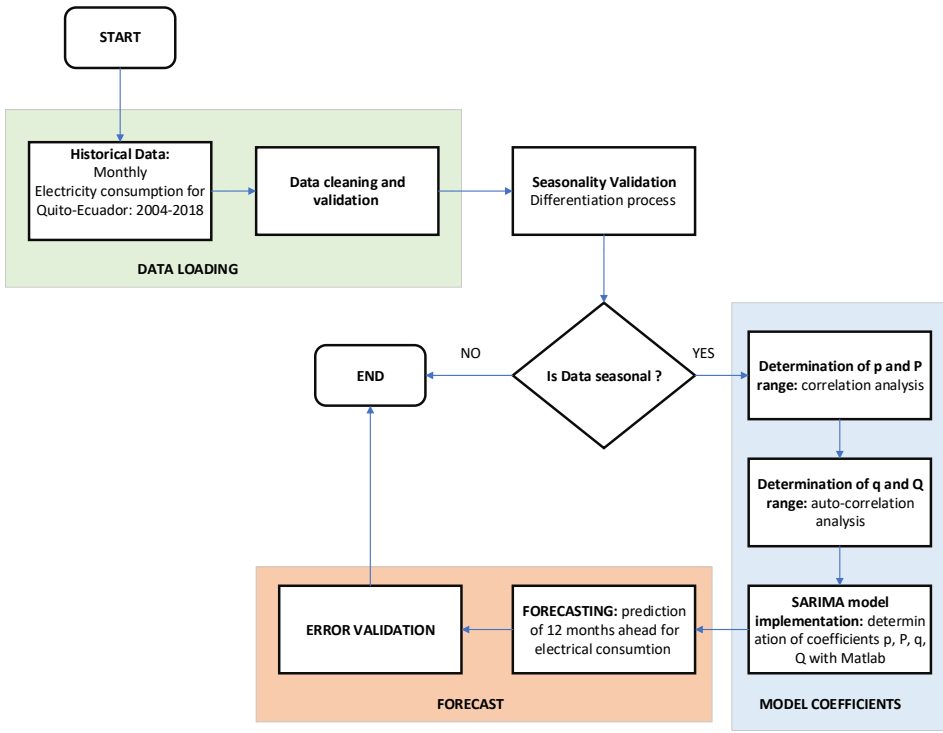

START
Historical Data: Monthly Electricity consumption for Quito-Ecuador: 2004-2018
Data cleaning and validation
Seasonality Validation Differentiation process
DATA LOADING
END
NO
Is Data seasonal ?
YES
Determination of p and P range: correlation analysis
Determination of q and Q range: auto-correlation analysis
SARIMA model implementation: determination of coefficients p, P, q, Q with Matlab
FORECASTING: prediction of 12 months ahead for electrical consumtion
ERROR VALIDATION
FORECAST
MODEL COEFFICIENTS

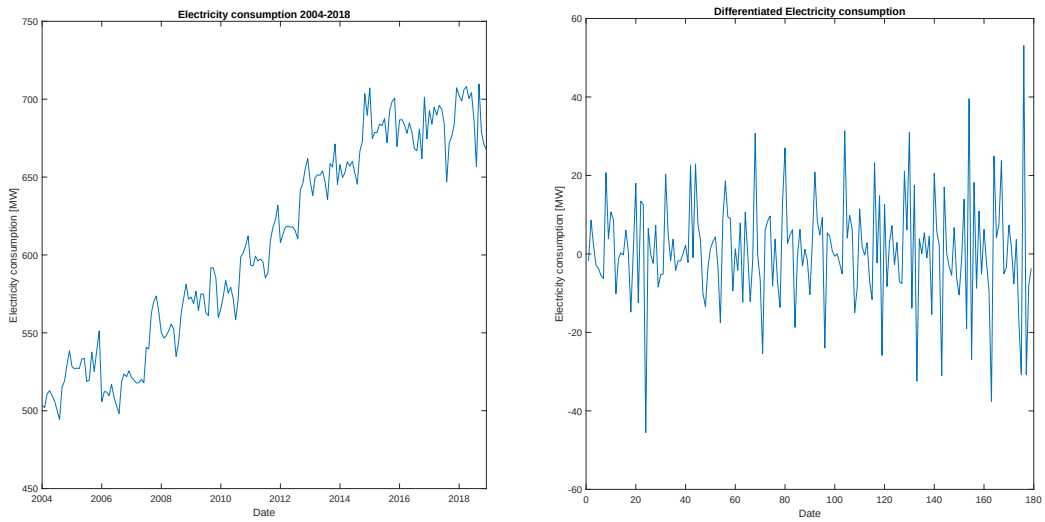
Figure 6.5: Methodology for electricity forecasting

any seasonal trends. If such movements are present, the time series of electricity consumption ought to show no increasing pattern post-differentiation. This entire procedure is conducted on Matlab; the corresponding code is shown below.

```
clc, clear
%% DATA LOADING
opts = detectImportOptions('DATA_MONTHLY_18.csv');
opts = setvaropts(opts,"DATE",'InputFormat','MM/dd/uuuu');
ec = readtable('DATA_MONTHLY_18.csv',opts);

demand=ec{:,"MAXDEMAND_MW_"};
date=ec{:,"DATE"};

%% VISUAL REPRESENTATION
% Original Data
figure()
subplot(2,1,1)
plot(date,demand)
title('Electricity consumption 2004-2018')
xlabel('Date')
ylabel('Electricity consumption [MW]')

%% SEASONALITY VERIFICATION
% Data differentiation
```

```
demand_dif=diff(demand);
subplot(2,1,2)
plot(demand_dif)
title('Differentiated Electricity consumption')
xlabel('Date')
ylabel('Electricity consumption [MW]')

h=gcf;
set(h,'PaperPositionMode','auto');
set(h,'PaperType','A4');
set(h,'PaperOrientation','landscape');
set(h,'Position',[10 0 500 800]);
set(h, 'InvertHardcopy', 'off')
figure = gcf;
figure.Color = 'white';
print -pdf -r800 figure1
```

After applying this analysis to the electricity consumption of Quito-Ecuador from 2004 to 2018, Figure 6.6 shows that this data corresponds to a seasonal time series.



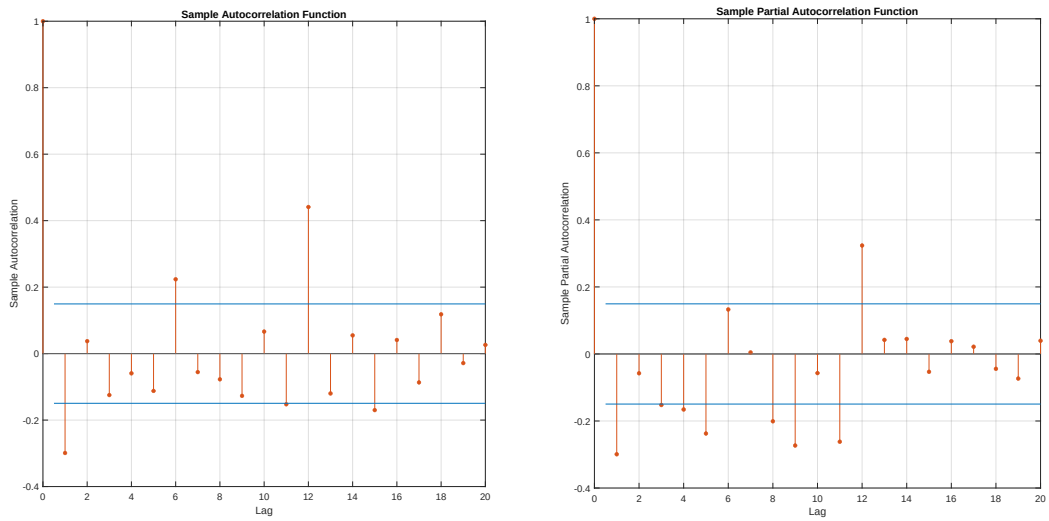
(a) Time series for electricity consumption (b) Differentiated time series

Figure 6.6: analysis for seasonality in electricity consumption for Quito-Ecuador from 2004 to 2018

Then, an analysis was performed on electricity time series data to ascertain the order of autoregressive and moving average coefficients for the SARIMA model. The data were scrutinized using correlation and autocorrelation techniques to identify significant lags greater than one. It helped to determine the order of each coefficient accurately and efficiently. The Matlab code for this analysis is shown as follows.

```
%% SERIES CORRELATION
figure()
autocorrect(demand_dif)
h=gcf;
set(h,'PaperPositionMode','auto');
set(h,'PaperType','A4');
set(h,'PaperOrientation','landscape');
set(h,'Position',[10 0 800 800]);
set(h, 'InvertHardcopy', 'off')
figure = gcf;
figure.Color = 'white';
print -pdf -r800 figure3

%% SERIES PARTIAL AUTOCORRELATION
figure()
parcorr(demand_dif)
h=gcf;
set(h,'PaperPositionMode','auto');
set(h,'PaperType','A4');
set(h,'PaperOrientation','landscape');
set(h,'Position',[10 0 800 800]);
set(h, 'InvertHardcopy', 'off')
figure = gcf;
figure.Color = 'white';
print -pdf -r800 figure4
```

Matlab analysis of correlation and partial auto-correlation shown in figure 6.7 reveals four significant lags in the time series correlation; this suggests that the autoregressive component should have an order ranging from 0 to 4. The partial auto-correlation analysis indicates seven significant lags, implying that the moving average order should range from 0 to 7.



(a) Correlation analysis for electricity consumption

(b) Partial auto-Correlation analysis for electricity consumption

Figure 6.7: analysis for autoregressive and moving average indexes order

So far, the analysis for electricity consumption has provided the following criteria:

- Autoregressive component p varies from 0 to 4.
- Moving average component q varies from 0 to 7.
- Considering that the time series has been differentiated once, $d = 1$ and $D = 1$.
- P takes the same values as p.
- Q takes the same values as q.
- As data was taken by one sample per month, seasonality $S = 12$

Next, it is necessary to build a SARIMA model in Matlab using the already discussed structure and determining the values for the associated coefficients (p, q, P, Q). Once the model is created, the error will be evaluated using equation 6.2.

```
%% SARIMA MODEL CREATION

%% Seasonal ARIMA Model
% Estimate a SARIMA Model of demand1
SARIMA_mdl = arima('Constant',0,'ARLags',1:4,...
    'D',1,'MALags',1:7,'SARLags',[12,24],...
    'Seasonality',0, 'SMALags',[12,24],...
    'Distribution', 'Gaussian');
SARIMA_mdl = estimate(SARIMA_mdl,demand,'Display','off');
```

```
10
11 % Sarima Model
12 residuals = infer(SARIMA_mdl,demand);
13 model = demand+residuals;
14
15 % RSS calculation for error
16 error_ab=demand - model;
17 error_p=error_ab./demand;
18 error_p=abs(error_p*100);
19 error_mean=mean(error_p)
20 RSS=sum(error_p)
21
22 figure()
23 plot(demand,'LineWidth',2)
24 hold on
25 plot(model,'--', 'LineWidth',2, 'Color', '#D95319')
26 legend('Original electricity consumption',...
27     'Time series created with SARIMA model,' 'Location', northwest')
28 title('Original time series vs Sarima Model')
29 xlabel('Sample')
30 ylabel('Electricity consumption [MW]')
31
32 h=gcf;
33 set(h,'PaperPositionMode','auto');
34 set(h,'PaperType','A4');
35 set(h,'PaperOrientation','landscape');
36 set(h,'Position',[10 0 1000 800]);
37 set(h, 'InvertHardcopy', 'off')
38 figure = gcf;
39 figure.Color = 'white';
40 print -pdf -r800 figure4
```

The values for each coefficient (p, q, P, Q), the residual sum of squares RSS and average error per sample are shown in table 6.1

Table 6.1: SARIMA model coefficients and results

Name	Description
p coefficients	[0.0848375, -1.12611, 0.128075, -0.931196] at lags [1 2 3 4]]
q coefficients	[-0.497183, 1.12716, -0.647343, 1, -0.546107, -0.0214771, -0.0948195] at lags [1 2 3 4 5 6 7]
P coefficients	[0.0265442, 0.806656] at lags [12 24]
Q coefficients	[0.293264, -0.504022] at lags [12 24]
Seasonal coefficient	$S = 12$
Residual sum of squares RSS	201.7595
Average error per sample	1.1209%

Finally, after creating the $SARIMA$ model and verifying that the RSS and average per sample error are acceptable, the next step consists of creating forecasted values for the next 12 months of electricity consumption. The Matlab code for this analysis is shown as follows.

```
1 %% FORECASTING DATA
2 % Plot only 8 last years
3 demand2=demand(100:end)
4 T=length(demand2);
```

```
[yF,yMSE] = forecast(SARIMA_mdl,12,demand);
upper = yF + 1.96*sqrt(yMSE);
lower = yF - 1.96*sqrt(yMSE);

aux=demand2(end-1:end);
yF=cat(1,aux,yF);
upper = cat(1,aux,upper);
lower = cat(1,aux,lower);


figure()
plot(demand2,'LineWidth',2)
hold on
h1 = plot(T-1:T+12,yF,' k', 'LineWidth',2, 'Marker', 'diamond');
h2 = plot(T-1:T+12,upper,'r--,' 'LineWidth,' 1.5,'Marker,''*');
plot(T-1:T+12,lower,' r--', 'LineWidth',1.5, 'Marker','*')
grid on
grid minor
xlim([0,T+11])
title('Electricity consumption forecast and 95% Forecast Interval')
xlabel('Sample')
ylabel('Electricity consumption [MW]')
legend([h1,h2], 'Forecast', '95 % Interval', 'Location','NorthWest')
hold off

h=gcf;
set(h,'PaperPositionMode','auto');
set(h,'PaperType','A4');
set(h,'PaperOrientation','landscape');
set(h,'Position',[10 0 1200 800]);
set(h, 'InvertHardcopy', 'off')
figure = gcf;
figure.Color = 'white';
print -pdf -r800 figure5
```

Forecasting values are shown in figure 6.8 in the black colored line. Also, this figure shows confidence intervals of $\pm 5\%$.

6.5 Conclusions

Energy forecasting plays a critical role in predicting future energy demand and ensuring the reliability of the power system. It allows utilities to plan for energy needs, manage supply, and identify potential vulnerabilities in the system. Accurate forecasting becomes even more crucial as the shift towards renewable energy sources continues due to their inherent variability. Therefore, investing in reliable and precise energy forecasting methods is imperative for utilities to meet customer demands and maintain a dependable power system.

By utilizing time series analysis, it is possible to accurately model and understand electricity consumption behavior. By creating a precise mathematical model, predictions can be made with a high degree of reliability. Research has shown that

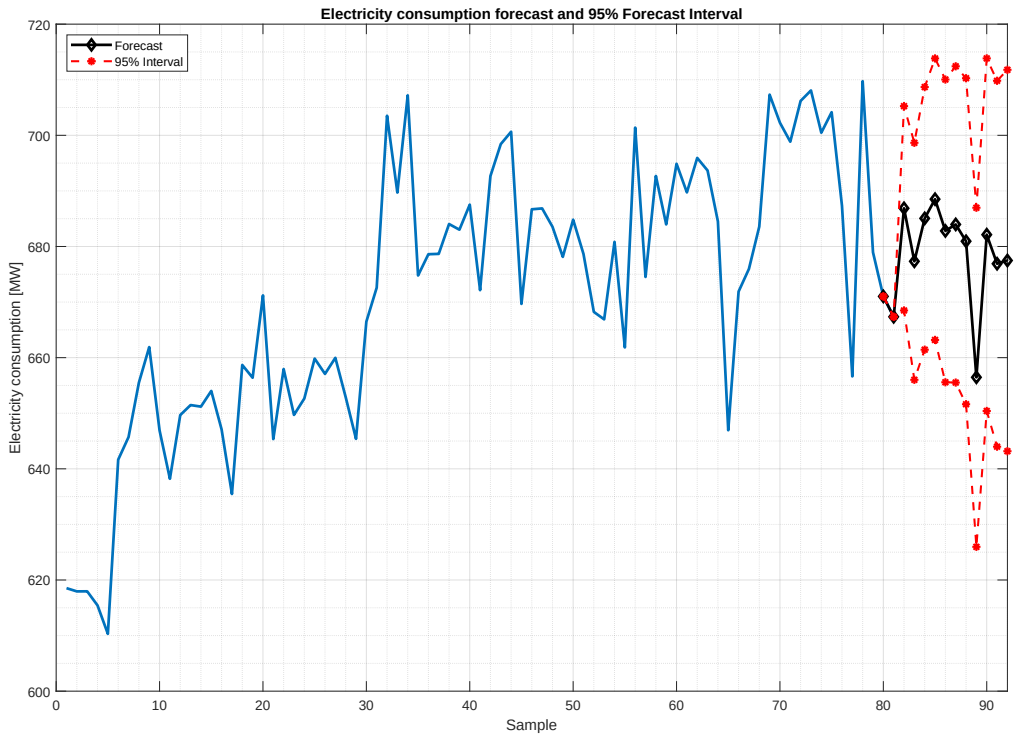


Figure 6.8: Electricity consumption forecast with 95% confidence intervals

such models can produce a small residual sum of squares (RSS) and average error per sample, further bolstering the dependability of predicted values.

In this work, a highly useful tool is presented that enables users to create an effective time series model for energy consumption. It should be noted that this tool is versatile and can be applied to any problem. It is crucial to assess the model's errors in order to determine its alignment with actual data. The residual sum of squares (RSS) is a key metric for this evaluation, as the closer it is to zero, the more accurate the predicted values will be. This tool is an excellent resource for anyone seeking to make reliable forecasts and optimize their energy usage.

Additionally, time series analysis can identify potential risks and opportunities in the energy market, allowing companies to adjust their strategies accordingly. Overall, time series analysis is an essential tool for energy companies looking to stay ahead of the curve and make informed decisions about the future of energy consumption.

Bibliography

[1] M. Jaramillo and D. Carrión, "An Adaptive Strategy for Medium-Term Electricity Consumption Forecasting for Highly Unpredictable Scenarios: Case Study Quito, Ecuador during the Two First Years of COVID-19", *Energies*, vol. 15, no. 22, pp. 1–18, 2022, ISSN: 19961073. DOI: 10.3390/en15228380.

[2] H. Bian, Y. Zhong, J. Sun, and F. Shi, "Study on power consumption load forecast based on K-means clustering and FCM–BP model", *Energy Reports*, vol. 6, pp. 693–700, 2020, ISSN: 23524847. DOI: 10.1016/j.egyr.2020.11.148. [Online]. Available: https://doi.org/10.1016/j.egyr.2020.11.148.

[3] J. Moon, J. Park, E. Hwang, and S. Jun, "Forecasting power consumption for higher educational institutions based on machine learning", *Journal of Supercomputing*, vol. 74, no. 8, pp. 3778–3800, 2018, ISSN: 15730484. DOI: 10.1007/s11227-017-2022-x.

[4] I. A. Bonchuk and A. P. Shaposhnikov, "Assessment of the Calculation Results of a Mathematical Model for Operational Forecasting of the Amount of Active Power Consumption", *Power Technology and Engineering*, vol. 56, no. 1, pp. 107–114, 2022, ISSN: 15701468. DOI: 10.1007/s10749-023-01481-4.

[5] X. Xiong, X. Hu, and H. Guo, "A hybrid optimized grey seasonal variation index model improved by whale optimization algorithm for forecasting the residential electricity consumption", *Energy*, vol. 234, 2021, ISSN: 03605442. DOI: 10.1016/j.energy.2021.121127.

[6] H. Kaur and S. Ahuja, "SARIMA modelling for forecasting the electricity consumption of a health care building", *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 12, pp. 2795–2799, 2019, ISSN: 22783075. DOI: 10.35940/ijitee.L2575.1081219.

[7] C. Hora, F. C. Dan, G. Bendea, and C. Secui, "Residential Short-Term Load Forecasting during Atypical Consumption Behavior", *Energies*, vol. 15, no. 1, 2022, ISSN: 19961073. DOI: 10.3390/en15010291.

[8] S. Li and Q. Wang, "India's dependence on foreign oil will exceed 90% around 2025 - The forecasting results based on two hybridized NMGM-ARIMA and NMGM-BP models", *Journal of Cleaner Production*, vol. 232, pp. 137–153, 2019, ISSN: 09596526. DOI: 10.1016/j.jclepro.2019.05.314. [Online]. Available: https://doi.org/10.1016/j.jclepro.2019.05.314.

[9] E. Chodakowska, J. Nazarko, and Ł. Nazarko, "Arima models in electrical load forecasting and their robustness to noise", *Energies*, vol. 14, no. 23, 2021, ISSN: 19961073. DOI: 10.3390/en14237952.

[10] D. Chi, "Research on electricity consumption forecasting model based on wavelet transform and multi-layer LSTM model", *Energy Reports*, vol. 8, pp. 220–228, 2022, ISSN: 23524847. DOI: 10.1016/j.egyr.2022.01.169. [Online]. Available: https://doi.org/10.1016/j.egyr.2022.01.169.

[11] W. Jiang, X. Wu, Y. Gong, W. Yu, and X. Zhong, "Holt–Winters smoothing enhanced by fruit fly optimization algorithm to forecast monthly electricity consumption", *Energy*, vol. 193, p. 116 779, 2020, ISSN: 03605442. DOI: 10.1016/j.energy.2019.116779. [Online]. Available: https://doi.org/10.1016/j.energy.2019.116779.

[12] M. D. Jaramillo, D. F. Carrión, and J. P. Muñoz, "A Novel Methodology for Strengthening Stability in Electrical Power Systems by Considering Fast Voltage Stability Index under N − 1 Scenarios", *Energies*, vol. 16, no. 8, 2023, ISSN: 1996-1073. DOI: 10.3390/en16083396. [Online]. Available: https://www.mdpi.com/1996-1073/16/8/3396.

[13] M. Jaramillo, D. Carrión, and J. Muñoz, "A Deep Neural Network as a Strategy for Optimal Sizing and Location of Reactive Compensation Considering Power Consumption Uncertainties", *Energies*, vol. 15, no. 24, 2022, ISSN: 1996-1073. DOI: 10.3390/en15249367. [Online]. Available: https://www.mdpi.com/1996-1073/15/24/9367.